在线零售站点的自适应和商业智能的发现

Adaptive Online Retail Web Site and Discovering Internet Marketing Intelligence

王 实[1] 高 文[1] 郎金文[2] 李锦涛[1]

(中国科学院计算技术研究所 北京100080)[1](郑州大学工学院 郑州450002)[2]

Abstract There are two important problems in online retail: 1) The conflict between the different interest of all customers to the different commodities and the commodity classification structure of Web site; 2)Many customers will simultaneously buy both the beer and the diaper that are classified in different classes and levels in the Web site, which is the typical problem in data mining. The two problems will make majority customers access overabundant Web pages. To solve these problems, we mine the Web page data, server data, and marketing data to build an adaptive model. In this model, the frequently purchased commodities and their association commodity sets that are discovered by the association rule discovery will be put into the suitable Web page according to the placing method and the backing off method. At last the navigation Web pages become the navigation content Web pages. The Web site can be adaptive according to the users' access and purchase information. In online retail, the designers require to understand the latent users' interest in order to convert the latent users to purchase users. In this paper, we give the approach to discover the Internet marketing intelligence through OLAP in order to help the designers to improve their service.

Keywords Web mining, Self-adaptation, Marketing intelligence

1 引言

电子商务已经被称为是Internet最重要的应用之一，WWW正以其简单易用性赢得越来越多的用户，为用户和商家提供了双向交流、"虚拟"交易的理想空间。在电子商务环境下，一个联机零售商在Web上开展电子商务的业务模型如图1[1]。其中市场数据存储商品信息和用户的交易信息；Web结构数据存储Web页和Web的结构。服务数据存储访问日志。

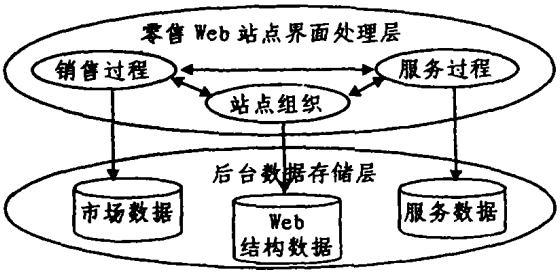

图1 在线零售模型

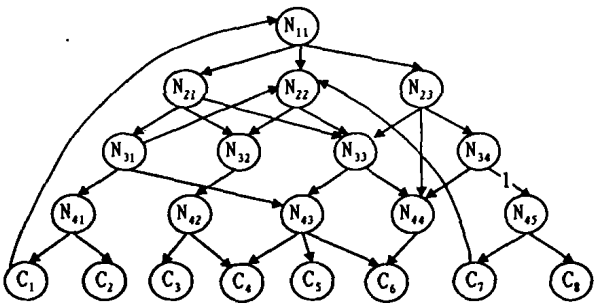

图2 在线零售站点Web站点分类结构模型

一个标准的在线零售网站的分类设计结构如图2所示。其中：每一个节点表示一个页面。N节点表示导航页或分类页；C节点表示内容页或购买页。网站的结构是介于树形和网状层次结构之间的一种结构。网站的设计者，会尽量考虑到达一个购买页存在多个路径；而且从物品的分类结构上说，很多物品属于多个类，如电子书，既属于书籍类又属于电子产品。规模越大的站点，其结构越复杂。

开展在线零售业务的一个主要问题就是：用户面对厂家提供的大量产品信息，不知如何有效提取；而厂家面对大量的用户，不知他们的兴趣和要求所在，因而不知如何调整其服务方式和产品结构：

1. 全体用户对零售商品的兴趣不一致，对物品的兴趣存在着一个概率分布，即全体用户对某些物品的兴趣要远远大于另一些物品。但Web站点Web页面结构的分类层次设计必须严格遵循商品的分类结构，因为如果不是这样，一般用户就无法访问。于是这两者之间就存在一种矛盾。这种矛盾所导致的结果是大量用户不得不浏览许多不相关的页面，进入Web站点的很多层次最终才能找到自己所需要的商品。解决这个问题的一种思路是将图1上的导航页(N页)变成导航内容页(NC页)。这样用户就可以在N页上直接购买自己需要的商品。

2. 许多用户购买的物品类似于啤酒和尿布这样的物品——属于数据挖掘中的经典问题，即在页面结构分类上两者相距很远，但很多顾客会同时购买，于是这些用户就不得不反复进入退出多个Web页，来完成购买。对于这样具有关联购买的物品集，要做的就是如何自动发现关联物品集，并且自动建立包含它们的导航内容页，以帮助用户访问。

所以需要建立一个模型和相应的算法，在各导航页上标注购买物品快捷清单。即经过算法处理后，N页要自动变成NC页，即导航购买页。原有的N页之间的导航关系不被破坏。NC页将满足大部分用户的需求，使他们不需访问过多的层次，或尽量不需绕路而进行购买。

解决这个问题的方法是：根据在线零售站点的Web访问结构、群体用户的访问日志和群体用户的交易记录，建立网站的自适应模型。在模型中，对频繁被购买商品通过搜索算法找

到推荐点，通过关联规则发现算法，从用户的交易数据中，发现关联购买集合；在推荐点上标注这些商品及其关联购买集合；当处理完所有的关联购买集合后，通过竞争来决定出现在导航页面上的物品集，最终将导航页合理地变成导航购买页。这样全体用户对这样的站点进行访问，他们的总的访问遍历数就会减小。于是通过这样的过程，这个站点就可以自动根据用户的访问购买情况，进行自适应。

文[1]首次提出将数据挖掘的技术应用于电子商务的环境下，以发现市场智能。挖掘的对象不仅包括日志、Web 页面，也包括市场数据。文[1]还给出了在电子商务环境下，挖掘的一个总的框架。但他们的方法依然局限在传统的挖掘手段。本文所述方法是建立在其基础之上，更好地发现并应用了市场智能。不仅应用传统的挖掘手段（如关联规则发现算法[2]）而且新定义和建立了自适应模型进行 Web 站点结构的有限智能调整，即使站点能够根据群体用户的访问而自适应。

文[3]首次给出 Web 挖掘的定义，并且给出一个关于 Web 访问信息挖掘的系统 WEBMINER。文中提到的思路是通过对 Web 站点的日志进行处理[4,5]，将数据组织成传统的数据挖掘方法能够处理的事务数据形式，然后利用传统的数据挖掘方法（如传统的关联规则发现算法）进行处理，其得出的挖掘结果也是传统的数据挖掘结果，并没有根据挖掘的结果调整站点的组织结构。我们的方法是在这种方法的基础上，不但得到挖掘的结果而且把挖掘的结果用于改进在线零售站点的组织结构，以利于群体用户的访问。

WebLogMiner[6]方法用 OLAP 技术来实现对 Web 日志数据的预测、分类、时间序列分析。其分析的结果没有用于 Web 站点的重新设计，而我们的方法是要主要用于站点的自动重新设计，且不破坏原有的分类结构，即自适应。

在文[7,8]中，这些方法的目的是自动定制不同的用户访问界面，其特点是：1）Web 站点或代理动态地把一些增强的当前可视的 Web 页面给用户，即定制个性化的页面；2）页面上的信息针对的是基于某种模型而得到的特定的某一个，或某一类用户；3）该模型基于该用户或该类用户以前的访问方式。对比来说，我们的方法：1）是一种优化方法；2）周期性、离线地进行挖掘；3）挖掘的对象是全体用户的交互行为，挖掘的是全体用户的共同访问购买兴趣，挖掘的结果面向全体用户；4）不需要特定的某一个或某一类用户的信息。

文[9]用聚类方法实现 Web 站点对外部访问的自适应。它通过 PageGather 聚类方法的结果：索引页，来帮助用户进行访问。这种方法使用聚类挖掘，在一个 Web 站点上寻找相关页面的集合，这种相关页面集合是根据总体用户的相关访问来决定的。采用的手段是创立相似矩阵，矩阵的元素是根据访问日志所得出的页面之间的共同被访问的频度，然后在这个矩阵中寻找每一个聚类，根据每一个聚类创立一个索引页。这种方法自适应的基本元素是每个 Web 页面，而且算法需要建立大量的索引页；而在本文方法中，自适应的基本元素是商品，方法本身不破坏 Web 站点原有的分类结构，不会形成附加的索引页，被提升的内容，自然而然地出现在它们应该出现的地方。

在零售业务中，客户对商品的兴趣和对商品的购买是两个不同的概念。在传统的零售业务中，只能记录对商品的购买信息，而无法直接得到用户对商品的兴趣信息。而对比传统的零售业务在在线零售站点中，用户访问信息可以记录更为详细的客户对商品的访问，因而通过挖掘可以得到用户对商品的兴趣知识。一旦服务方了解到顾客的兴趣知识和购买知识，那么就可以采用相应的商业促销手段把潜在的用户转变为实际的用户。本文给出一些关于商业智能的定义以帮助服务方发现用户的浏览兴趣，并给出相应的 OLAP 方法以发现这些商业智能，以帮助服务方更好地开展商业服务。和文[1]所述方法相比，本文方法将用户的访问兴趣与用户的购买行为分开，利用 OLAP 技术进行挖掘。

2 数据准备

2.1 挖掘对象

挖掘的对象存在于图1所示的后台数据存储层之中。具体分为：

1）用户的访问日志（在图1的服务数据中）。服务器上的日志格式遵循 W3C 标准[13]；

表1 用户访问日志

Field	Description
Date	Date, time, and timezone of request
Client IP	Remote host IP and / or DNS entry
User name	Remote log name of the user
Bytes	Bytes transferred (sent and received)
Server	Server name, IP address and port
Request	URI query and stem
Status	http status code returned to the client
Service name	Requested service name
Time taken	Time taken for transaction to complete
Protocol version	Version of used transfer protocol
User agent	Service provider
Cookie	Cookie ID
Referrer	Previous page
…	…

2）用户的交易纪录即传统的交易事务数据（在图1的市场数据中）。交易数据记录用户对物品的购买信息。

表2 用户交易记录

Field	Description
TransactionID	Transaction Identification
Commodities	The Commodities that the user purchases in the transaction

3）Web 页面的结构信息（在图1的 Web 结构数据中）。

2.2 生成用户访问事务

进行挖掘时，首先要将一段时间用户的访问日志组织成用户访问事务数据。设 L 为用户访问日志，其中的一个项 $l\in L$ 包括用户的 IP 地址 $l.ip$，用户的标识符 $l.uid$，被存取页的 URL 地址 $l.url$，以及存取访问的时间 $l.time$。访问事务被定义为：

$$t=\langle ip_t, uid_t, \{(l'_1.url, l'_1.time), \cdots, (l'_m.url, l'_m.time)\}\rangle$$
$$where, for\ 1\leqslant k\leqslant m, l'_k\in L, l'_k.ip=ip_t, l'_k.uid=uid_t, l'_k.time-l'_{k-1}.time\leqslant C \quad (1)$$

这里 C 是一个固定的时间窗。

对 Log 进行处理，找到每一个事务，然后就可以对这些事务进行关联规则发现。寻找访问事务的算法为：

1. 对日志进行预处理。

2. 根据每一个访问者 IP，划分日志。即在 Log 中找到每

一个访问者的访问记录集。

3. 对每一访问者的访问记录集，根据 C 进行分割，找到每一个访问者的每一次访问记录集，这时，每一个访问者的每一次访问记录集就构成了一个访问事务。

4. 最终按时间排序的所有访问事务构成我们进行挖掘的基础。

处理完日志后我们就有了用户访问事务集 T；此时，对 T 进行处理，把用户每次发生购买行为和由此而进行的路径访问提取出来，形成用户访问购买事务集。其中每一条记录不仅包括每一个用户的交易记录，而且也包括该用户发生一次购买时，他对 Web 站点的访问记录，即他的存取路径：

表3 用户访问购买记录

Field	Description
TransactionID	Transaction Identification
Commodities	The Commodities that the user purchases in the transaction
AccessPath	The user went through the Web path for buying the goods

用户访问购买事务集构成我们建立自适应模型的基础。

3 在线零售站点的自适应模型

在线零售自适应模型基于如下四种基本元素：

1)物品：为用户购买的目标。

2)页面：页面本身分为导航页、购买页、导航购买页。每个购买页面，或导航购买页面包含一些物品，一个物品可能出现在多个购买页面，或导航购买页面页面中。

3)页面之间的层次关系：这种层次关系反映出用户购买一个物品至少需要访问的页面的个数。如果一个常被购买的物品处于较低的层次，那么需要将其提升以减少这些用户的访问层次数。

4)用户的访问：通过对用户访问状况的挖掘，可以得到群体用户购买一个物品时所通过的每一个页面的次数。

3.1 基本定义

为了建立在线零售站点的自适应模型，首先需要给出一些基本定义：

定义1 一个用户访问购买事务 tp：由于在在线零售站点内，每个页面内包含一组物品，那么一个用户访问购买事务是由一个用户所访问的页面和其在该页面上购买的物品构成：

$$tp = \langle (url^1, pruchaseset^1), (url^2, pruchaseset^2), \cdots, (url^m, pruchaseset^m) \rangle \quad (2)$$

其中，集合 $purchase^i$ 为用户在 url^i 页面上购买的物品集。那么设定 TP 为全部用户访问购买事务的集合。

定义2 群体用户对物品 σ_j 购买时，通过 url_i 页面的次数 $Count(\sigma_j|url_i)$；一个物品的购买都存在着一个购买次数 $Count(\sigma_j|q_i)$：

$$Count(\sigma_j|url_i) = \|\{tp|tp\in TP, \sigma_j\in purchaseset^r, url_i = url^r, 1\leqslant s\leqslant r\leqslant m\}\| \quad (3)$$

定义3 在线零售站点的自适应模型 Adaptive Site Model：一个在线零售站点的自适应模型 ASM 是基于所定义的群体用户迁移模型，经过如下扩展而得出的：

1)一个状态集合 Q，具有指定的初始状态 q_{start} 和最终状态 q_{end}。每个状态表征为被用户访问的一个 Web 页面。

2)两个状态之间存在一条有向边，对应于群体用户在这两个页面之间的转移。

3)存在物品集 $\sum = \sigma_1, \sigma_2, \cdots, \sigma_M$。每个状态包含该物品集的一个子集 $\sum_q = \sigma_1^q, \sigma_2^q, \cdots, \sigma_M^q$。

4)在每一个节点 q_i 上，群体用户对物品集 $\sum$ 中的每一个物品的购买都存在着一个通过次数 $Count(\sigma_1|q_i), \cdots, Count(\sigma_M|q_i)$。

5)自适应过程。通过自适应过程，可以使得经常被用户购买的物品被提升，不常被购买的物品下降，物品分布的有限自调整。所谓有限是指调整的过程依然遵循物品的总的分类结构。

根据第2节的内容以及式(2)和式(3)和随后所述的自适应过程，就可以建立起这样一个模型。

由于每个状态节点的空间有限，设每个状态节点只能放置 θ 个物品。

定义4 在 q_i 节点上已放置物品的集合 $Marked(q_i)$：$Marked(q_i)$ 为在 q_i 节点经过放置算法计算后，决定被放置的物品的集合。$Marked(q_i)$ 集合中物品的个数为 θ 个。

定义5 放置 σ_j 物品的节点的集合 $Position(\sigma_j)$：经过放置算法计算后，已决定放置 σ_j 物品的节点的集合。

定义6 σ_j 物品在 q_i 节点的得益：

$$profit(\sigma_j|q_i) = Count(\sigma_j|q_i) - Count(\sigma_j|q_i \wedge q_m)$$
$$where: q_m \in Position(\sigma_j) \quad (4)$$

其中 $Count(\sigma_j|q_i, \{q_m\})$ 表示为在 TP 中同时出现 σ_j 和 q_i 和 q_m 记录的个数。在 q_m 节点为任意一个已决定放置 σ_j 物品的节点。

3.2 自适应过程

模型的自适应过程为：

1)从初始状态 q_{start} 开始对自适应模型进行分层遍历。q_{start} 为第 L_0 层，从 q_{start} 开始到那些直接可达的状态为第 L_1 层，等等。

2)从第 L_1 层开始逐层向下在每个状态节点上根据放置策略，标注具有较大购买概率的那些物品。

3)放置策略：放置时，根据如下策略决定是否放置：

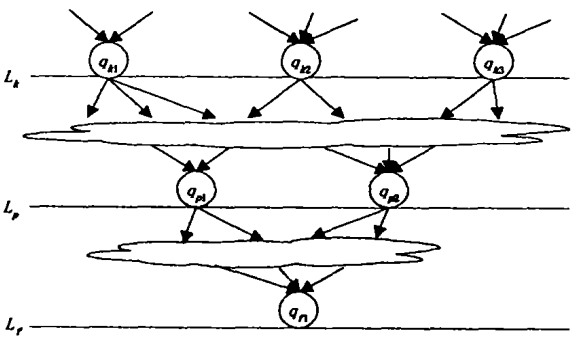

图3 放置状况示意图

如图3所示，如果在 f 层以上均已放置完毕，开始在 f 层上的 q_{f1} 节点进行放置，设该节点购买数不为0的物品数为 n，那么放置策略为：

$$\|Marked(q_{f1})\| = \theta,$$
$$for\ each\ \sigma_T^* \in Marked(q_{f1}), each\ \sigma_k^i \in Candidate(q_{f1}),$$
$$profit(\sigma_T^*|q_{f1}) \geqslant profit(\sigma_k^i|q_{f1}),$$
$$profit(\sigma_1^*|q_{f1}) \geqslant profit(\sigma_2^*|q_{f1}) \geqslant \Lambda \geqslant profit(\sigma_\theta^*|q_{f1}), profit(\sigma_1^i|q_{f1}) \geqslant profit(\sigma_2^i|q_{f1}) \geqslant \Lambda \geqslant profit(\sigma_{n-\theta}^i|q_{f1}) \quad (5)$$

其中，$Candidate(q_{f1})$为未被放置的物品的集合。

该策略实际上反映的是一种竞争的关系，在任意一个节点上，如果一个物品的购买数较大，但如果该物品在上层的一些节点上已被放置，在该节点上的得益较小；而另外一个物品的购买数虽较小，但如果该物品在上层的一些节点上未被放置，在该节点上的得益较大，那么前一个物品可能竞争不过后一个物品，即只有后一个物品才会被放置在该节点上。

4）后退策略：在放置策略中，竞争过程是面向全部在该节点的物品和群体用户通过该节点而购买的物品。如果原有的在该节点的物品没有竞争到有限的位置，那么就要考虑放置到下层，参加下层的下一次竞争。后退时需要退到与其语义最相近的分类结构中。

3.3 自适应算法

相应的自适应算法为

```
Algorithm: AdaptiveSite
Input: ASM, W, E /* ASM 为自适应模型，Count(σj|url1)已根据公式3计算得出；W 为分割矩阵，行为每一状态，列为每一种物品，其表示每个页面放置哪些物品，其初始都为0值，如果一个页面被决定放置一个物品，那么相应的值为1；E 结构同于 W，表示现有的哪个页面正在放置哪些物品 */
Begin:
  LevelSearch(ASM);
  /* 从 qstart 节点开始对模型进行逐层遍历，得 L0, L1, …, Lt 层 */
  j:= 0;
  While j≤t do begin
    For each q∈Lj do begin
      MarkedSetj = GetMarkedSet(ASM, W, q); /* 根据公式4和5计算在 q 状态上的被放置元素集。*/
      ModifyW(W, MarkedSet, q); /* 在 W 矩阵中 q 所在的行上将 MarkedSet 的每一个元素置1 */
      FallBack(ASM, q, W, E); /* 如果原有的在 q 上的物品竞争失败，那么就要退到下一层相应的状态上 */
    End For;
    j:=j+1;
  End While;
End.
Output: W
```

3.4 放置关联购买集合

设定支持度和可信度，通过对交易事务数据采用关联规则发现算法，可以得到关联规则集合。在自适应算法结束后，为了便捷用户的购买，在每个状态点上，不仅标注竞争获胜的物品集，而且标注这些物品的关联购买集。这样一来例如当用户购买完“啤酒”之后就可以直接在该页面购买“尿布”。

4 系统总的处理框架

整个系统按图4所示结构进行处理：

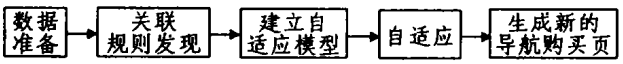

图4 系统总的处理框架

在一个 T 时间段内，经过群体用户的访问，会得到新的挖掘对象，然后执行如下处理过程：

1）数据准备：根据用户访问记录和用户交易记录得到用户访问购买事务集 TP。

2）关联规则发现：在 TP 中寻找关联物品集。

3）建立自适应模型：根据用户的访问购买事务集 TP 和原有站点的拓扑结构，在每一个节点 q_i 上，计算 σ_j 物品的通过次数 $Count(\sigma_j|q_i)$。

4）自适应：利用自适应算法在模型上进行自适应以得到在每个节点上的放置物品集。

5）生成新的导航购买页：在每个节点上根据放置物品集和关联物品集以及原有页面的拓扑结构生成新的导航购买页。

5 商业智能的发现

对在线零售电子商务站点而言，最重要的一个任务就是发现潜在的购买用户的兴趣，以开展相应的活动进行促销，把潜在的用户转换为实际的购买用户。

对比常规的商业智能发现，由于访问日志（记录了用户的访问行为，等价于在常规商业智能发现中能够记录下用户所看的各个柜台），交易日志的存在，那么在在线零售电子商务站点上，可以挖掘出比常规商业事务数据更精细的商业智能。

5.1 基本定义

当用户在在线零售站点进行访问时，由于其经过一些页面，并且可能在一些页面发生购买行为，那么对公式2所定义用户访问购买事务给出如下扩展定义：

$$tap=\langle Url, ObserverSet, PurchaseSet\rangle$$
$$Url=\{url^1, url^2, \Lambda, url^m\}$$
$$ObserverSet=\{\sigma_j|\sigma_j \text{ is in } url^i\}$$
$$PurchaseSet=purchaseset^1\cup purchaseset^2\cup\cdots\cup purchaseset^m \quad (6)$$

那么相应的 TAP 表征全体用户访问购买事务的集合。设定 TAP 共有 n 个 tap 事务。

通过定义一些商业智能规则，那么就可以设计相应的算法用于发现商业智能：

定义7 对 σ_j 物品的观察支持度 $observersupport(\sigma_j)$：其含义为用户对 σ_j 物品的观察次数。

$$observersupport(\sigma_j)=\|\{tap|tap\in TAP, \sigma_j\in ObserverSet\}\| \quad (7)$$

定义8 对 σ_j 物品的购买支持度 $purchasesupport(\sigma_j)$：其含义为用户对 σ_j 物品的购买次数。

$$purchasesupport(\sigma_j)=\|\{tap|tap\in TAP, \sigma_j\in PurchaseSet\}\| \quad (8)$$

定义9 一个通过 url_i 的用户访问关键字时长 $length^u(url_i,\sigma)$：对一个用户而言，如果他对某个概念感兴趣，那么他会访问具有该概念的页面，而且会较长时间地访问具有该概念的页面。设用户对一个页面 url_i 的访问时长为 $Length^u(url_i)$，如果该页面具有 f 个物品 $\sigma_1, \sigma_2, \cdots, \sigma_f$，那么设定该用户在该页面对每个物品的访问时间长度为：

$$length^u(url_i,\sigma_j)=\frac{Length^u(url_i)}{f}, j=1,\cdots,f \quad (9)$$

那么在一个通过 url_i 的用户访问购买事务 $t^u(url_i)$ 中，用户通过 url_i 对一个物品 σ_j 访问的总时长 $sum^u(url_i,\sigma_j)$ 为：

$$sum^u(url_i,\sigma_j)=\sum_{i=j}^{m} length^u(url_i,\sigma_j) \quad (10)$$

定义10 对 σ_j 物品的观察时间支持度 $observertimesupport(\sigma_j)$：群体用户对 σ_j 物品总的观察时间被定义为对 σ_j 物品的观察时间支持度：

$$observertimesupport(\sigma_j)=\sum_{u=1}^{n} sum^u(q_{start},\sigma_j) \quad (11)$$

定义11 对 σ_j 物品的购买信心度 $PurchaseConfidence(\sigma_j)$：

$$PurchaseConfidence(\sigma_j)=\frac{observersupport(\sigma_j)}{purchasesupport(\sigma_j)} \quad (12)$$

购买信心度表征对一个给定的物品 σ_j 物品，顾客平均观看多少次才会进行一次购买。

定义12 对 σ_j 物品的购买时间信心度 $PurchaseConfidencetime(\sigma_j)$：

$$PurchaseConfidencetime(\sigma_j)=\frac{observertimesupport(\sigma_j)}{purchasesupport(\sigma_j)} \quad (13)$$

购买时间信心度表征对一个给定的物品 σ_i 物品，顾客平均观看多长时间才会进行一次购买。

定义13 顾客满意度 $SatisfactionConfidence$：

$$LargeObserverItemSet=\{\sigma\mid observersupport(\sigma)\geqslant\theta_1\}$$

$$LargePurchaseItemSet=\{\sigma\mid purchasesupport(\sigma)\geqslant\theta_2\}$$

$$SatisfactionCnfidence=\frac{\|LargeObserverItemSet\cap LargePurchaseItemSet\|}{\|LargeObserverItemSet\cup LargePurchaseItemSet\|} \quad (14)$$

其中，θ_1，θ_2为两个阈值。该指标说明如下问题：顾客最想买的东西是设为 A 集合。B 集合为顾客实际最常买的物品的集合。如果两个集合的交集中元素的个数与两个集合的并集中元素的个数的比值越大，那么就说明顾客常看的与常买的较为一致，否则就需要考虑更新营销策略。顾客满意度这一指标能够反映出顾客对网站的设计结构、促销手段等好坏的评价值。

5.2 发现商业智能

为了发现上述商业智能，给出如下关于用户访问购买事务六维数据超立方体：

定义14 用户访问购买事务数据超立方体 $UAPC$：一个 $UAPC$ 表征了一个6维信息空间：

$$UAPC=[D_1,D_2,D_3,D_4,D_5,D_6]$$

其中每一个 D_i 表征了 $UAPC$ 的一个维。

定义15 $UAPC$ 的一个维 D_i：一个维 D_i 表征 m 个属性，例如 $D_i=[a_{i1},a_{i2},\cdots,a_{im}]$。其中一些属性可以不具有实际意义，仅为了聚集数据的目的而存在。

那么在该数据超立方体中，总的单元的个数为：

$$NumberofUnit=\prod_{i=1}^{6}\|UAPC_i\| \quad (15)$$

其中$\|UAPC_i\|$为 $UAPC$ 中第 i 维的属性的个数。

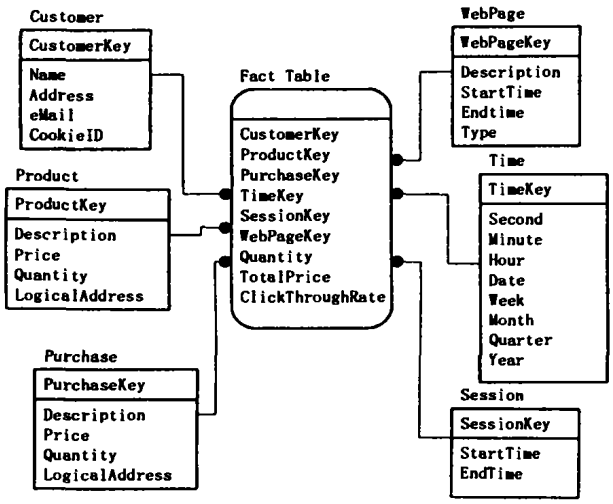

图5 用户访问购买事务数据超立方体基础上的数据库星形模式

定义16 在 $UAPC$ 的一个维 D_i 上存在的一个概念层次 G_i：G_i 为一个有向、连结、无环图：

$$G_i=(D_i,E_i),$$

$$where\ D_i=\{a_{i1},a_{i2},\cdots,a_{im}\},E_i=\{e_{i1},e_{i2},\cdots,e_{ik}\} \quad (16)$$

其中，a_{ij}为 D_i 的属性值，每个 e_{if}为一条边 $e_{if}=(a_{ip},a_{iq})$，a_{ip}，$a_{iq}\in D_i$，$p<q$ 表示 a_{ip}是 a_{iq}的一个子概念。a_{i1}的入度为零。如果 $a_{ip},a_{iq}\in D_i$，$p<q$，且 a_{ip}是 a_{iq}的一个子概念那么就存在一条边 $e_{if}=(a_{ip},a_{iq})$。

在这个数据超立方体的基础上，给定一个星形模式(图5)以用于 OLAP 分析。

给定数据立方体和相应的星形模式，那么就可以利用各种商业 OLAP 工具来进行分析以得到相应的支持度、信心度和满意度。

6 实验

实验分为两步：首先在模型阶段(通过建立一个以图2为背景的13个物品的简单站点例子，运行了一段时间以得到数据)；其次在相当于真实站点环境下进行了实验。

6.1 实验比较

以图2为例，表4给出 $Count(\sigma_j|q_i)$。

表4 在每个节点对每种物品的通过次数 $Count(\sigma_j|q_i)$

	C_1		C_2	C_3		C_4			C_5	C_6		C_7	C_8
	G_1	G_2	G_3	G_4	G_6	G_5	G_7	G_8	G_9	G_{10}	G_{11}	G_{12}	G_{13}
N_{11}	30	10	10	30	30	10	10	30	20	40	60	10	10
N_{21}	30	10	10	30	0	10	0	0	0	2	0	0	0
N_{22}	0	0	0	0	30	0	10	30	0	20	0	0	0
N_{23}	0	0	0	0	0	0	0	0	20	0	60	10	10
N_{31}	30	10	10	0	0	0	0	0	0	1	0	0	0
N_{32}	0	0	0	30	30	10	0	0	0	0	0	0	0
N_{33}	0	0	0	0	0	0	10	30	20	30	30	0	0
N_{34}	0	0	0	0	0	0	0	0	0	0	0	10	10
N_{41}	30	10	10	0	0	0	0	0	0	0	0	0	0
N_{42}	0	0	0	30	30	10	0	0	0	0	0	0	0
N_{43}	0	0	0	0	0	0	10	30	20	30	10	0	0
N_{44}	0	0	0	0	0	0	0	0	0	10	20	0	0
N_{45}	0	0	0	0	0	0	0	0	0	0	0	10	10

设定 $\theta=1$，那么经过自适应算法运算后结果如下：

表5 自适应算法的运算结果

N_{11}	N_{21}	N_{22}	N_{23}	N_{31}	N_{32}	N_{33}	N_{34}	N_{41}	N_{42}	N_{43}	N_{44}	N_{45}
G_{11}	G_1	G_6	G_{10}	G_2	G_4	G_8	G_{12}	G_3	G_5	G_9	G_9	G_{13}

简单统计方法主要有以下两种方式，这两种方式应用于非树形的站点结构，都存在着严重的缺点：

1)从一个导航节点 q 开始，它所能到达的所有的内容节点中，那些具有最大被购买概率 $P(G|q_{start})$(在首节点的被购买概率，即物品在根节点的被购买概率)的物品，被作为内容部分放到该导航节点上。

2)从一个导航节点 q 开始，它所能到达的所有的内容节点中，那些具有最大被购买概率 $P(G|q)$(在该节点的被购买概率，即物品在该节点的被购买概率)的物品，被作为内容部分放到该导航节点上。

第一种方法的缺点在于：以 N_{31}为例，在该节点按照该方法应该放入 G_{11}。但群体用户到达该节点的主要购买目的是 G_1而不是 G_{11}。而第二种方法是对第一种方法的改进。

如果站点的结构是复杂的网状层次结构，如图2所示，那么当物品在导航站点竞争时，第二种方法又会遇到如下问题：

设每一个导航页的内容部分只能存放一个物品，那么根据简单统计方法：N_{42}节点就会放入 G_4或 G_6物品，但是因为 G_4或 G_6物品已分别在 N_{22}和 N_{32}节点中放入，这就产生冗余放置的问题，即要买 G_4或 G_6物品已分别在 N_{22}和 N_{32}节点中买到，那么在节点上应该剔除在上层节点中已购买的元素的

影响，即本文所述的方法就是要解决这个问题。

6.2 具有真实背景的实验

实验环境为具有 Intel Xeon500 双 CPU，512M 内存，80G 硬盘，Windows NT 操作系统的服务器。

我们选取了中科院计算所的 Web 服务器(http://www.ict.ac.cn)上的日志作为实验对象，实验数据包括从1998年11月到1999年11月用户对计算所 Web 站点一年的访问数据。整个站点包括352个 html 页面。用户访问日志的总量为：147M，包括1749934项。经过事务识别算法，共识别出10399个用户访问事务，平均访问事务长度为8.8，即用户平均每次访问8.8个页面。实验定义了11个关键字：$\sum$ = {人物，机构，论文，学报，论坛，新闻，人才培养，招聘，简介，科研，挂靠单位}；根据每个页面的内容，在各个页面上分别标注 $\sum$ 的子集，以每个用户访问事务的最后一个节点上的关键字作为被购买的物品，以产生用户访问购买事务集 TP。表6给出一些放置的结果(θ=2)。图6给出算法的运行时间。

表6 ICT 数据集中一些页面的放置结果

页面	放置集
/	“科研”，“人物”
/kkk/nn25.htm	“学报”，“人物”
/kkk/nn.htm	“简介”，“人才培养”
/luntan.htm	“论坛”，“新闻”
/chpc/	“论文”，“机构”

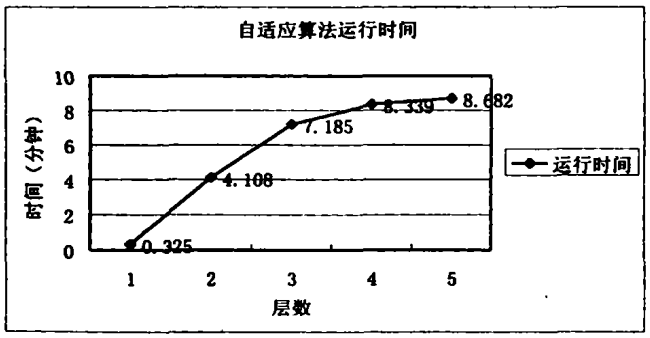

图6 ICT 数据集中自适应算法的运行时间

7 结论以及将来的工作

开展在线零售业务存在的问题是群体用户必须浏览许多不相关的页面，才能最终找到自己所需要的商品。为了解决该问题，本文建立一个在线零售站点的自适应模型。在在线零售站点中，服务方需要了解用户的浏览兴趣以把潜在的用户转变为实际的用户。本文给出相应的利用 OLAP 发现商业智能的方法，以帮助服务方更好地开展有针对性的服务。

本文所述的自适应方法本质上是 Web 访问信息挖掘中(Web Usage Mining)的一种推荐方法，即根据群体用户对在线零售电子商务站点的访问，在 Web 站点上推荐根据对以前群体用户的访问兴趣挖掘而得到的知识，以加速当前群体用户对站点的访问效率。在该方法中，建立模型的训练过程较为简单，相应的各个公式比较容易计算。该工作朝着建立完全自适应的 Web 站点作出了贡献：

1)分析了在线零售站点，用户访问时存在的冗余访问问题，并给出了解决这种问题的办法，即建立网站的自适应模型。

2)在模型中，对频繁被购买商品通过放置策略和后退策略找到推荐点，通过关联规则发现算法发现关联购买集合；在推荐点上标注这些商品及其关联购买集合；最终将导航页合理地变成导航购买页，即站点可以自动根据群体用户的访问购买情况进行自适应。

3)在模型中，关联规则发现算法被合理地结合起来，可以发现并且解决在在线零售电子商务站点上依然存在的啤酒和尿布的问题。而且利用 Web 站点的优点，可以更容易地解决这个问题。

4)该方法完全是自动的，不需要人工的干预。

自适应模型方法的特点是：1)本质上是一种优化方法；2)周期性、离线地进行挖掘；3)挖掘的对象是全体用户的购买行为，挖掘的是全体用户的共同访问兴趣，挖掘的结果面向全体用户；不需要特定的某一个或某一类用户的信息；4)方法在本质上是自动跨不同物品分类集的；5)挖掘的结果是不影响现存的分类结构；6)建立导购页的过程，等价于数据挖掘的结果进行可视化处理；7)这种方法不影响以后的挖掘；8)自适应的粒度不是页面而是更深入到了物品。

在本文所述商业智能发现方法中，可以发现在传统的零售业务中，无法直接得到用户对商品的兴趣信息所表征的兴趣知识。那么结合传统的购买知识的发现，一旦服务方了解这些知识，那么就可以采用相应的商业促销手段把潜在的用户转变为实际的用户。本文给出一些关于商业智能的定义以帮助服务方发现用户的浏览兴趣，并给出相应的 OLAP 方法以发现这些商业智能，帮助服务方更好地开展商业服务。

我们进一步的工作将不仅是 Web 访问信息挖掘中的推荐方法而且是预测方法。通过将这两种方法结合起来，我们不但能够在 Web 站点上推荐我们所发现的用户的兴趣，而且也将能够预测用户的兴趣。

参考文献

1 Buchner A G, Mulvenna M D. Discovering Internet Marketing Intelligence through Online Analytical Web Usage Mining. SIGMOD Record, 1998, 27(4)

2 Agrawal R, Srikant R. Fast algorithms for mining association rules. In: Proc. of the 20th VLDB Conf. Santiago, Chile, 1994. 487～499

3 Cooley R, Mobasher B, et al. Web Mining: Information and Pattern Discovery on the World Wide Web. In: Proc. of Intl. Conf. on Tools with Artificial Intelligence. Newport Beach, IEEE, 1997

4 Cooley R, Mobasher B, et al. Grouping Web Page References into Transactions for Mining World Wide Web Browsing Patterns. In: Proc. of Knowledge and Data Engineering Workshop, Newport Beach, CA, IEEE. 1997

5 Cooley R, Mobasher B, et al. Data Preparation for Mining World Wide Web Browsing Patterns. Knowledge and Information Systems, 1999, 1(1)

6 Zaiane O R, Xin M, et al. Discovering Web Access Patterns and Trends by Applying OLAP and Data Mining Technology on Web Logs. In: Proc. of Advances in Digital Libraries, Santa Barbara, CA, 1998

7 Joachims T, Freitag D, Michell T. Web-watcher: A tour guide for the world wide web. In: Proc. 15th Int. Joint Conf. AI, 1997. 770～775

(下转第38页)

针对个人信息空间的有限点来选择数据，对于每一个点选择出来的数据都是确切反应该用户群的交易数据。基于这些更能反映该用户群且数据量更小的数据，采用与4.1节相同的方法来进行数据挖掘。由用户群交易信息挖掘出来的相关商品将符合该用户群的消费水平，例如对其他个性化信息一样，收入较高的人群，关联模式将提供价格稍高且商品质量较好的同一类商品。

这样将有效促进每一用户群数据挖掘算法的计算效率，而且挖掘出来的关联模式将更准确地反映该用户群的相关商品信息。根据不同的用户群建立不同的挖掘模式索引，这样当输入某一个特定用户群时，相应的索引将迅速挖掘相应中粒度级个性化信息。

由上面的分析可以看出，个人信息空间中点的多少与数据计算量是一种彼此消长的关系。即个人信息空间中的点越多，挖掘计算量越小，同时由于分类太多，类的维护开销增大，存储空间增大。寻找分类的多少与计算量大小的最佳平衡点是一个新的研究课题(在此不作详细讨论)，其输入是网站交易信息量的大小，平衡点是多方面因素综合后的一个优化问题。

4.3 ECPI Miner 的设计

ECPI Miner 是一个电子商务个性化信息挖掘工具。它将关联分析的算法集成于其中，输入是用户的个人信息(一般用户或有个人注册信息的某一用户群)和此次购买商品，输出是根据挖掘算法计算出来的相关商品。如图2。

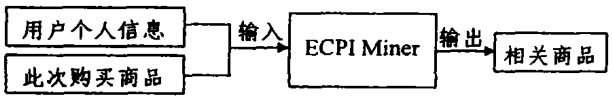

图2 ECPI Miner 工作方式

结论 利用数据挖掘的方法实现了两种不同级别的粗粒度个性化服务。即针对有个人注册信息的会员，可提供基于用户群粒度级的个性化服务。针对一般的非注册用户提供广粒度级的个性化服务，即根据所有的交易信息来计算相关商品的提供模式。

参考文献

1 Li Changqing, Tang Shiwei, Li Hongyan, Zhao Wenbing. Improving on the Personalized Algorithm and Research on the Pertinence of Influence Index in E-Business. In: Proc. of the 9th National Conf. on Multimedia Technology NCMT'2000. Oct. 2000. 285～288

2 Tian Shengfeng, Huang Houkuan. Artificial Intelligence and Knowledge Engineering. China Railway Press, 1999. 8

3 Wang Shan, et al. Data Warehouse Technology and OLAP. Science Presss, 1998. 6

4 Hu Kan, Xia Shaowei. Large Data Warehouse-based Data Mining: a Survery. Journal of Software, 1998, (1): 53～63

5 Mannila H, Tovivonen H, Inkeri Verkamo A. Efficient Algorithm for Discovering Association Rules. In: Proc. of AAAI Workshop on Knowledge Discovery in Database. July 1994. 181～192

6 Srikant R, Agrawal R. Mining Generalized Association Rules. In: Proc. of the 21th Intl. Conf. on Very Large Database. Sept. 1995. 407～419

7 Toivonen H, Klemettinen M, Ronkaine P, et al. Pruning and Grouping Discovered Association Rules. Mlnet Workshop on Statistics, Machine Learning and Discovery in Database. Heraklion, Crete, Greece, April 1995

8 Agrawal R, Srikant R. Fast Algorithm for Mining Association Rules. In: Proc. of the 21th Intl. Conf. on Very Large Database. Santiago, Chile, Sept. 1994

(上接第35页)

8 Fink J, Kobsa A, Nill A. User-oriented Adaptivity and Adaptability in the AVANTI Project. In Designing for the Web: Empirical Studies. 1996

9 Perkowitz M, Etzioni O. Adaptive Web Sites: Automatically Synthesizing Web Pages. In: Proc. of AAAI98. 1998

10 Perkowitz M, Etzioni O. Adaptive web sites: an AI challenge. In: Proc. 15th Int. Joint Conf. AI. 1997a

11 Perkowitz M, Etzioni O. Adaptive web sites: Automatically learning from user access patterns. In: Proc. of the Sixth Int. WWW conference. 5. 1997b

12 Rabiner L R. A tutorial on hidden Markov models and selected applications in speech recognition. Proceedings of the IEEE, 1989, 77(2): 257～286

13 Luotonen A. The common log file format. http://www.w3.org/pub/WWW/, 1995

14 Zaiane O R. Resource and knowledge discovery from the internet and multimedia repositories, doctoral dissertation, university of Simon Fraser, 1999

15 Mladenic D. Machine Learning on non-homogeneous, distributed text data, doctoral dissertation, university of Ljubljana, 1998

16 Chen M S, Park J S, Yu P S. Data mining for path traversal patterns in a web environment. In ICDCS, 1996. 385～392

17 Brin S, Page L. The anatomy of a large-scale hypertextual web search engine. In: 7th Int. Conf. WWW, Brisbane, Australia, April 1998

18 Chakrabarti S, et al. Experiments in topic distillation. In: ACM SIGIR Workshop on Hypertext Information Retrieval on the Web, Melbourne, Australia, 1998

19 Wexelblat A, Maes P. Footprints: History-rich web browsing. In: Proc. Conf. Computer-Assisted Information Retrieval (RIAO), 1997. 75～84

20 Spiliopoulou M, Faulstich L C. WUM: A Web Utilization Miner. 1998

21 Spiliopoulou M. The laborious way from data mining to web mining. Int. Journal of Comp. Sys., Sci. & Eng., Special Issue on "Semantics of the Web". Mar. 1999

22 Craven M, et al. Learning to Extract Symbolic Knowledge from the World Wide Web. In: Proc. of the 15th National Conf. on Artificial Intelligence (AAAI-98). 1998