(19)中华人民共和国国家知识产权局

(12)发明专利

(10)授权公告号 CN 101320375 B
(45)授权公告日 2010.09.22

(21)申请号 200810063010.4

(22)申请日 2008.07.04

(73)专利权人 浙江大学
地址 310027 浙江省杭州市浙大路 38 号

(72)发明人 吴江琴 庄越挺 袁川 张寅

(74)专利代理机构 杭州求是专利事务所有限公司 33200
代理人 张法高

(51)Int.Cl.
G06F 17/30(2006.01)

审查员 施鹏韬

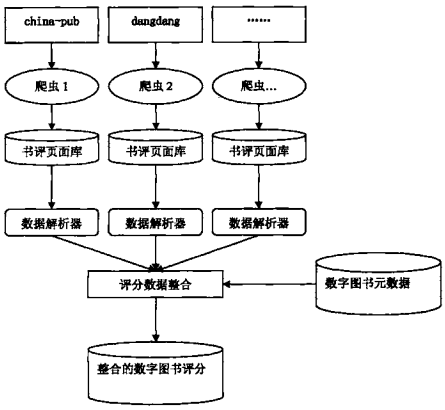
权利要求书 2 页 说明书 5 页 附图 1 页

(54)发明名称

基于用户点击行为的数字图书搜索方法

(57)摘要

本发明公开了一种基于用户点击行为的数字图书搜索的方法。首先，提取Web日志中的图书阅读记录构建图书之间的关联图，使用该关联图来计算图书的相关性排序；其次，提取日志中的检索阅读记录，利用其中读者对检索结果的隐式反馈对查询词进行聚类；最后，在查询词聚类的基础之上，针对每类查询词，利用读者对检索结果的隐式反馈，综合从关联图得出的图书相关性排序、互联网上的图书评分以及文本相似度这三种信息源，形成最终的图书搜索结果排序。本发明可以获得客观的图书相关性排序和很好的查询词聚类效果；将互联网上丰富的图书评分数据融入到相对封闭和静态的数字图书馆中，有效提高图书搜索质量。



CN 101320375 B

1. 一种基于用户点击行为的数字图书搜索方法，其特征在于包括以下步骤：

(1) 提取日志中的图书阅读记录构建图书之间的关联图，使用关联图计算图书的相关性排序得分；

(2) 提取日志中的检索阅读记录，利用读者对检索结果的隐式反馈对查询词进行聚类；

(3) 抓取互联网上的图书评分数据，整合形成图书评分排序得分；

(4) 在查询词聚类的基础之上，针对每类查询词，利用读者对检索结果的隐式反馈，综合从关联图得出的图书相关性排序、互联网上的图书评分以及文本相似度这三种排序信息源，形成最终的图书搜索结果排序；

所述的提取日志中的图书阅读记录构建图书之间的关联图，使用关联图计算图书的相关性排序得分步骤：分析数字图书馆的 Web 使用日志数据，提取 Web 使用日志数据中的读者图书阅读记录，构建共同阅读过图书 i 和图书 j 的读者的数量矩阵$\tilde{C}_{i,j}$，用 $U = \{u_i : 0 \leqslant i < m\}$ 表示读者的集合，$B = \{b_j : 0 \leqslant j < n\}$ 表示图书的集合，然后对$\tilde{C}_{i,j}$进行归一化处理：

$$C_{i,j} = \frac{\tilde{C}_{i,j}}{w_j} \text{ 其中 : } w_j = \sum_{0 \leq i < |B|} \tilde{C}_{i,j}$$

得到图书关联矩阵 $C_{i,j}$，使用向量 $BR = [br_0, br_1, br_2, \ldots, br_{|B|-1}]^T$ 表示图书的相关性排序得分，向量 $d = [d_0, d_1, d_2, \ldots, d_{|B|-1}]^T$ 表示图书的已知质量信息，最后使用下面的迭代方法来计算最终的图书相关性排序得分：

$$\begin{cases} BR(0) = \dfrac{1}{|B|} 1_{|B|} \\ BR(n+1) = \alpha CBR(n) + (1-\alpha) d \end{cases}$$

其中，α 为随机跳转概率，取值范围在 0 到 1 之间；

所述的提取日志中的检索阅读记录，利用读者对检索结果的隐式反馈对查询词进行聚类步骤：分析数字图书馆的Web使用日志数据，提取Web使用日志数据中的图书检索阅读记录，得到与每个查询词相关的数字图书，使用 $Q = \{q_i : 0 \leqslant i < m\}$ 表示查询词的集合，集合 $B_i \subseteq B$ 表示与查询词 q_i 相关的图书，集合 $G_{i,j}$ 定义为：

$$G_{i,j} = \begin{cases} B_i \cap B_j & \text{if } i \neq j \\ \varnothing & \text{if } i = j \end{cases}$$

构建表示与查询词 q_i 和 q_j 都相关的图书的数量的矩阵，定义为：

$$\tilde{S}_{i,j} = |G_{i,j}|$$

归一化后即得到查询词相似性矩阵：

$$S_{i,j} = \frac{\tilde{S}_{i,j}}{w_j} \text{ 其中 : } w_j = \sum_{0 \leq i < |Q|} \tilde{S}_{i,j}$$

然后使用类似上述的图书相关性排序的迭代计算方式来对查询词进行聚类；

所述的抓取互联网上的图书评分数据，整合形成图书评分排序得分步骤：设计爬虫程序抓取互联网上著名图书网站上的图书评分页面，分析提取图书的元数据和图书评分，如

果提取出来的图书在数字图书馆中存在，则整合不同网站上的相同图书的评分数据，设整合的站点为 S_i 其中：$0 \leqslant i < N$，图书 b_k 在站点 S_i 上的归一化后的评分数据为 v_{ki}，评分人数为 p_{ki}，若该图书该站点上不存在或存在但是没有评分记录，那么 v_{ki} 或 p_{ki} 为零，使用如下公式来整合图书评分：

$$w_k = \sum_{0 \leq i < N} \frac{v_{ki} p_{ki}}{\sum_{0 \leq i < N} p_{ki}} \quad ;$$

所述的在查询词聚类的基础之上，针对每类查询词，利用读者对检索结果的隐式反馈，综合从关联图得出的图书相关性排序、互联网上的图书评分以及文本相似度这三种排序信息源，形成最终的图书搜索结果排序步骤：将基于数字图书访问关联图的图书排序值表示为 $R = [r_0, r_1, r_2, \ldots, r_{|B|-1}]^T$，将从互联网上整合得到的图书排序值表示为 $S = [s_0, s_1, s_2, \ldots, s_{|B|-1}]^T$，将基于元数据文本相似度的检索得分表示为 $T = [t_0, t_1, t_2, \ldots, t_{n-1}]^T$，其中 B 为图书的集合，三个排序值都是介于 0 到 1 之间的浮点值，得分最高的图书的分值为 1，对于一次图书搜索，使用基于文本相似度的元数据检索获得匹配的图书列表 $B = [b_0, b_1, b_2, \ldots, b_{n-1}]^T$，$b_k$，$0 \leqslant k < n$ 为图书的编号，图书列表文本相似度得分为 $T = [t_0, t_1, t_2, \ldots, t_{n-1}]^T$，n 为匹配当前搜索关键词的图书数目，然后使用如下公式来计算最终的图书得分 f_k：

$$f_k = \alpha t_k + \beta r_{b_k} + \gamma s_{b_k}$$ 其中：$0 \leqslant k < n$

其中 α，β，γ 为实待估参数，按照如下方式确定：从读者的图书检索阅读的序列数据中提取出一系列的有序对 <key，book>，进而将有序对 <key，book> 转化为 <key，score>，score 表示读者对图书的评分，得到的一系列有序对 <key，score>，按照查询词的聚类结果来将前面得到的有序对分为 m 个组，第 i 组中的所有有序对满足 key ∈ Q_i，Q_i 表示查询词的集合，对于第 i 组中的每一个有序对 <key_j，$score_j$>，计算出以它的 key_j 值作为查询词的检索结果中它对应的图书的三个排序分值，以有序对 <key_j，$score_j$> 的 $score_j$ 作为最终的图书得分 f_{ij}，这样第 i 组中的所有有序对构成一个多元线性回归分析模型：

$$f_{ij} = \alpha_i t_{ij} + \beta_i r_{ib_j} + \gamma_i s_{ib_j}$$

使用每一组中的所有有序对来进行最小二乘估计可以得到针对每一个查询词类的 α_i，β_i，γ_i。

基于用户点击行为的数字图书搜索方法

技术领域

[0001] 本发明涉及数字图书馆、信息检索和 Web 使用挖掘领域，尤其涉及一种基于用户点击行为的数字图书搜索方法。

背景技术

[0002] 21 世纪是数字化的时代，随着计算机技术、海量存储技术和网络技术的飞速发展，信息载体的数字化和信息传播的网络化得到了空前的深化，图书馆的数字化成为一个必然趋势。数字图书馆在世界很多国家受到了高度关注，并取得了迅猛发展，已经成为人们获取信息与知识的重要途径。

[0003] 数字图书馆中通常拥有海量的数字图书资源，如何有效的利用这些丰富而宝贵的资源，让数字图书馆读者能够更充分的利用他们就显得非常重要。数字图书搜索是数字图书馆必须提供的支撑性服务本，它是数字图书馆中最为重要的一个功能模块，它使得读者能够很好地找到需要的图书资源，是数字图书馆服务平台的“第一线”。

[0004] 传统的图书资源搜索系统是基于关系数据库的简单匹配查找，只能过滤出与读者所输入的关键字相匹配的相关图书条目，并没有使用有效的图书排序机制以提高读者检索的满意度，而且性能和用户体验不佳。

发明内容

[0005] 本发明为克服数字图书馆中传统图书搜索系统质量差的缺点，提供了一种高质量的图书搜索结果排序方法。

[0006] 基于用户点击行为的数字图书搜索方法包括以下步骤：

[0007] (1) 提取日志中的图书阅读记录构建图书之间的关联图，使用关联图计算图书的相关性排序得分；

[0008] (2) 提取日志中的检索阅读记录，利用读者对检索结果的隐式反馈对查询词进行聚类；

[0009] (3) 抓取互联网上的图书评分数据，整合形成图书评分排序得分；

[0010] (4) 在查询词聚类的基础之上，针对每类查询词，利用读者对检索结果的隐式反馈，综合从关联图得出的图书相关性排序、互联网上的图书评分以及文本相似度这三种排序信息源，形成最终的图书搜索结果排序。

[0011] 所述的提取日志中的图书阅读记录构建图书之间的关联图，使用关联图计算图书的相关性排序得分步骤：分析数字图书馆的 Web 使用日志数据，提取 Web 使用日志数据中的读者图书阅读记录，构建共同阅读过图书 i 和图书 j 的读者的数量矩阵$\tilde{C}_{i,j}$，用 $U = \{u_i : 0 \leqslant i < m\}$ 表示读者的集合，$B = \{b_j : 0 \leqslant j < n\}$ 表示图书的集合，然后对$\tilde{C}_{i,j}$进行归一化处理：

[0012] $C_{i,j}=\frac{\tilde{C}_{i,j}}{w_j}$ 其中 $:w_j=\sum_{0\le i<|B|}\tilde{C}_{i,j}$

[0013] 得到图书关联矩阵 $C_{i,j}$，使用向量 $BR=[br_0,br_1,br_2,\ldots,br_{|B|-1}]^T$ 表示图书的相关性排序得分，向量 $d=[d_0,d_1,d_2,\ldots,d_{|B|-1}]^T$ 表示图书的已知质量信息，最后使用下面的迭代方法来计算最终的图书相关性排序得分：

[0014] $$\begin{cases} BR(0)=\frac{1}{|B|}1_{|B|} \\ BR(n+1)=\alpha CBR(n)+(1-\alpha)d \end{cases}$$

[0015] 其中，α 为随机跳转概率，取值范围在 0 到 1 之间。

[0016] 所述的提取日志中的检索阅读记录，利用读者对检索结果的隐式反馈对查询词进行聚类步骤：分析数字图书馆的 Web 使用日志数据，提取 Web 使用日志数据中的图书检索阅读记录，得到与每个查询词相关的数字图书，使用 $A=\{q_i:0\le i<m\}$ 表示查询词的集合，集合 $B_i\subseteq B$ 表示与查询词 q_i 相关的图书，集合 $G_{i,j}$ 定义为：

[0017] $$G_{i,j}=\begin{cases} B_i\cap B_j & \text{if } i\neq j \\ \varnothing & \text{if } i=j \end{cases}$$

[0018] 构建表示与查询词 q_i 和 q_j 都相关的图书的数量的矩阵，定义为：

[0019] $\tilde{S}_{i,j}=|G_{i,j}|$

[0020] 归一化后即得到查询词相似性矩阵：

[0021] $S_{i,j}=\frac{\tilde{S}_{i,j}}{w_j}$ 其中 $:w_j=\sum_{0\le i<|Q|}\tilde{S}_{i,j}$

[0022] 然后使用类似上述的图书相关性排序的迭代计算方式来对查询词进行聚类。

[0023] 所述的抓取互联网上的图书评分数据，整合形成图书评分排序得分步骤：设计爬虫程序抓取互联网上著名图书网站上的图书评分页面，分析提取图书的元数据和图书评分，如果提取出来的图书在数字图书馆中存在，则整合不同网站上的相同图书的评分数据，设整合的站点为 S_i 其中 $:0\le i<N$，图书 b_k 在站点 S_i 上的归一化后的评分数据为 v_{ki}，评分人数为 p_{ki}，若该图书该站点上不存在或存在但是没有评分记录，那么 v_{ki} 或 p_{ki} 为零，使用如下公式来整合图书评分：

[0024] $w_k=\sum_{0\le i<N}\frac{v_{ki}p_{ki}}{\sum_{0\le i<N}p_{ki}}$ 。

[0025] 所述的在查询词聚类的基础之上，针对每类查询词，利用读者对检索结果的隐式反馈，综合从关联图得出的图书相关性排序、互联网上的图书评分以及文本相似度这三种排序信息源，形成最终的图书搜索结果排序步骤：将基于数字图书访问关联图的图书排序值表示为 $R=[r_0,r_1,r_2,\ldots,r_{|B|-1}]^T$，将从互联网上整合得到的图书排序值表示为 $S=[s_0,s_1,s_2,\ldots,s_{|B|-1}]^T$，将基于元数据文本相似度的检索得分表示为 $T=[t_0,t_1,t_2,\ldots,t_{n-1}]^T$，其中 B 为图书的集合，三个排序值都是介于 0 到 1 之间的浮点值，得分最高的图书的分值为 1，对于一次图书搜索，使用基于文本相似度的元数据检索获得匹配的图书列表 $B=[b_0,b_1,$

$b_2,\ldots,b_{n-1}]^T$，b_k，$0 \leqslant k < n$ 为图书的编号，图书列表文本相似度得分为 $T = [t_0, t_1, t_2, \ldots, t_{n-1}]^T$，n 为匹配当前搜索关键词的图书数目，然后使用如下公式来计算最终的图书得分 f_k：

[0026] $$f_k = \alpha t_k + \beta r_{b_k} + \gamma s_{b_k}$$ 其中：$0 \leqslant k < n$

[0027] 其中 α，β，γ 为实待估参数，按照如下方式确定：从读者的图书检索阅读的序列数据中提取出一系列的有序对 <key，book>，进而将有序对转化为 <key，score>，score 表示读者对图书的评分，得到的一系列有序对 <key，score>，按照查询词的聚类结果来将前面得到的有序对分为 m 个组，第 i 组中的所有有序对满足 key ∈ Q_i，Q_i 表示查询词的集合，对于第 i 组中的每一个有序对 <key_j，$score_j$>，计算出以它的 key_j 值作为查询词的检索结果中它对应的图书的三个排序分值，以有序对 <key_j，$score_j$> 的 $score_j$ 作为最终的图书得分 f_{ij}，这样第 i 组中的所有有序对构成一个多元线性回归分析模型：

[0028] $$f_{ij} = \alpha_i t_{ij} + \beta_i r_{ib_j} + \gamma_i s_{ib_j}$$

[0029] 使用每一组中的所有有序对来进行最小二乘估计可以得到针对每一个查询词类的 α_i，β_i，γ_i。

[0030] 本发明所述的基于用户点击行为的数字图书搜索方法具有如下特点：

[0031] (1) 充分挖掘和使用了数字图书馆 Web 使用日志中的用户点击行为数据，可以获得客观的图书相关性排序和很好的查询词聚类效果；

[0032] (2) 将互联网上丰富的图书评分数据融入到相对封闭和静态的数字图书馆中，有效提高图书搜索质量；

[0033] (3) 综合考虑了三种排序信息源，并使用读者的隐式反馈来进行集成，可以得到高质量的排序结果。

附图说明

[0034] 图 1 是整合图书评分数据基本框架图；

具体实施方式

[0035] 基于用户点击行为的数字图书搜索方法包括如下步骤：

[0036] 1. 提取日志中的图书阅读记录构建图书之间的关联图，使用关联图计算图书的相关性排序得分

[0037] 设读者的集合为：$U = \{u_i : 0 \leqslant i < m\}$，其中 m 为读者的数量；图书的集合为：$B = \{b_j : 0 \leqslant j < n\}$，其中 n 为图书的数量。读者与图书的阅读关系表示为：$T = \{t_{i,j} : u_i \in U \wedge b_j \in B, 0 \leqslant i < m, 0 \leqslant j < n\}$，其中 $t_{i,j}$ 为一个 bool 值，如果读者 u_i 阅读了图书 b_j 的 20％以上的页数，则视为一次有效的阅读行为，$t_{i,j}$ 等于 True，否则视为没有发生过该阅读行为，$t_{i,j}$ 等于 False：

[0038]

$$t_{i,j} = \begin{cases} True & \text{if } u_i \text{有效阅读} b_j \\ False & \text{else} \end{cases}$$

[0039] 图书与图书之间存在着关联关系，本发明通过同时阅读过两本图书的读者把这两本书关联起来。定义 $U_{i,j} \subseteq U$，为 U 的一个子集，表示同时阅读过图书 b_i 和图书 b_j 的读者的

集合：

[0040]

$$U_{i,j}=\begin{cases}\{u_k:(t_{k,i}\in T\wedge t_{k,i}=True)\wedge(t_{k,j}\in T\wedge t_{k,j}=True)\} & \text{if } i\neq j\\ \varnothing & \text{if } i=j\end{cases}$$

[0041] 这样就可以构建一个 |B|*|B| 的矩阵$\tilde{C}_{i,j}$，表示每一对图书之间共同阅读过它们的读者的数量：

[0042] $\tilde{C}_{i,j}=|U_{i,j}|$

[0043] 这里｜•｜表示一个集合的势。很明显$\forall i,\tilde{C}_{i,i}=0$而且$\tilde{C}$是一个对称矩阵。

[0044] 对矩阵$\tilde{C}$做规格化处理：

[0045] $C_{i,j}=\frac{\tilde{C}_{i,j}}{w_j}$其中：$w_j=\sum_{0\le i<|B|}\tilde{C}_{i,j}$

[0046] 称 C 为图书关联矩阵，矩阵中的每一项代表了图书对的关联系数，图书关联矩阵也可以被看作是一个图书关联图 G_C 的加权关联矩阵。G_C 中的节点表示集合 B 中图书，图书 b_i 和图书 b_j 之间有边链接当且仅当 $C_{i,j}>0$，$C_{i,j}$ 为边的权重。有一点需要注意，虽然$\tilde{C}$是一个对称矩阵，但是 C 并不是一个对称矩阵，所以边 (b_i, b_j) 和边 (b_j, b_i) 的权重并不是一致的，G_C 是一个有向加权图。

[0047] 图书相关性排序算法最基本的思想就是通过从用户使用日志中提取出来的图书关联图来传播图书的质量信息，进而估算出图书的相关性排序。使用向量 $BR=[br_0, br_1, br_2, \ldots, br_{|B|-1}]^T$ 表示图书的相关性排序得分，向量 $d=[d_0, d_1, d_2, \ldots, d_{|B|-1}]^T$ 表示图书的已知质量信息，最后使用下面的迭代方法来计算最终的图书相关性排序得分：

[0048]

$$\begin{cases}BR(0)=\frac{1}{|B|}1_{|B|}\\ BR(n+1)=\alpha CBR(n)+(1-\alpha)d\end{cases}$$

[0049] 其中，α 为随机跳转概率，取值范围在 0 到 1 之间。

[0050] 收敛后得到的 BR 即为图书 rank 值，在最终排序检索结果中它是一个重要的组成部分。最后对得到的结果进行处理使得其中 rank 值最高的项值为 1：

[0051] $br_i=\frac{br_i}{\max(BR)}$ max(BR) 为 BR 中值最高的项。

[0052] 2. 提取日志中的检索阅读记录，利用读者对检索结果的隐式反馈对查询词进行聚类

[0053] 分析数字图书馆的 Web 使用日志数据，提取其中的图书检索阅读记录，得到与每个查询词相关的数字图书，使用 $Q=\{q_i:0\le i<m\}$ 表示查询词的集合，集合$B_i\subseteq B$表示与查询词 q_i 相关的图书，集合$B_i\subseteq B$表示与查询词 q_i 相关的图书，集合 $G_{i,j}$ 定义为：

[0054]

$$G_{i,j}=\begin{cases}B_i\cap B_j & \text{if } i\neq j\\ \varnothing & \text{if } i=j\end{cases}$$

[0055] 构建表示与查询词 q_i 和 q_j 都相关的图书的数量的矩阵，定义为：

[0056] $$\tilde{S}_{i,j}=|G_{i,j}|$$

[0057] 归一化后即得到查询词相似性矩阵：

[0058] $$S_{i,j}=\frac{\tilde{S}_{i,j}}{w_j}\text{其中：}w_j=\sum_{0\le i<|Q|}\tilde{S}_{i,j}$$

[0059] 然后使用类似上述的图书相关性排序的迭代计算方式来对查询词进行聚类。

[0060] 3. 抓取互联网上的图书评分数据，整合形成图书评分排序得分

[0061] 设计爬虫程序抓取互联网上著名图书网站上的图书评分页面，分析提取图书的元数据和图书评分，如果提取出来的图书在数字图书馆中存在，则整合不同网站上的相同图书的评分数据，设整合的站点为 S_i 其中：$0 \leqslant i < N$，图书 b_k 在站点 S_i 上的归一化后的评分数据为 v_{ki}，评分人数为 p_{ki}，若该图书该站点上不存在或存在但是没有评分记录，那么 v_{ki} 或 p_{ki} 为零，使用如下公式来整合图书评分：

[0062] $$w_k=\sum_{0\le i<N}\frac{v_{ki}p_{ki}}{\sum_{0\le i<N}p_{ki}}$$ 。

[0063] 4. 在查询词聚类的基础之上，针对每类查询词，利用读者对检索结果的隐式反馈，综合从关联图得出的图书相关性排序、互联网上的图书评分以及文本相似度这三种排序信息源，形成最终的图书搜索结果排序

[0064] 将基于数字图书访问关联图的图书排序值表示为 $R=[r_0, r_1, r_2, \ldots, r_{|B|-1}]^T$，将从互联网上整合得到的图书排序值表示为 $S=[s_0, s_1, s_2, \ldots, s_{|B|-1}]^T$，将基于元数据文本相似度的检索得分表示为 $T=[t_0, t_1, t_2, \ldots, t_{n-1}]^T$，其中 B 为图书的集合，三个排序值都是介于 0 到 1 之间的浮点值，得分最高的图书的分值为 1，对于一次图书搜索，使用基于文本相似度的元数据检索获得匹配的图书列表 $B=[b_0, b_1, b_2, \ldots, b_{n-1}]^T$，$b_k$，$0 \leqslant k < n$ 为图书的编号，图书列表文本相似度得分为 $T=[t_0, t_1, t_2, \ldots, t_{n-1}]^T$，n 为匹配当前搜索关键词的图书数目，然后使用如下公式来计算最终的图书得分 f_k：

[0065] $$f_k=\alpha t_k+\beta r_{b_k}+\gamma s_{b_k}\text{其中：}0 \leqslant k < n$$

[0066] 其中 α，β，γ 为实待估参数，按照如下方式确定：从读者的图书检索阅读的序列数据中提取出一系列的有序对 <key, book>，进而将有序对转化为 <key, score>，score 表示读者对图书的评分，得到的一系列有序对 <key, score>，按照查询词的聚类结果来将前面得到的有序对分为 m 个组，第 i 组中的所有有序对满足 $key \in Q_i$，Q_i 表示查询词的集合，对于第 i 组中的每一个有序对 $<key_j, score_j>$，计算出以它的 key_j 值作为查询词的检索结果中它对应的图书的三个排序分值，以有序对 $<key_j, score_j>$ 的 $score_j$ 作为最终的图书得分 f_{ij}，这样第 i 组中的所有有序对构成一个多元线性回归分析模型：

[0067] $$f_{ij}=\alpha_i t_{ij}+\beta_i r_{ib_j}+\gamma_i s_{ib_j}$$

[0068] 使用每一组中的所有有序对来进行最小二乘估计可以得到针对每一个查询词类的 α_i，β_i，γ_i。

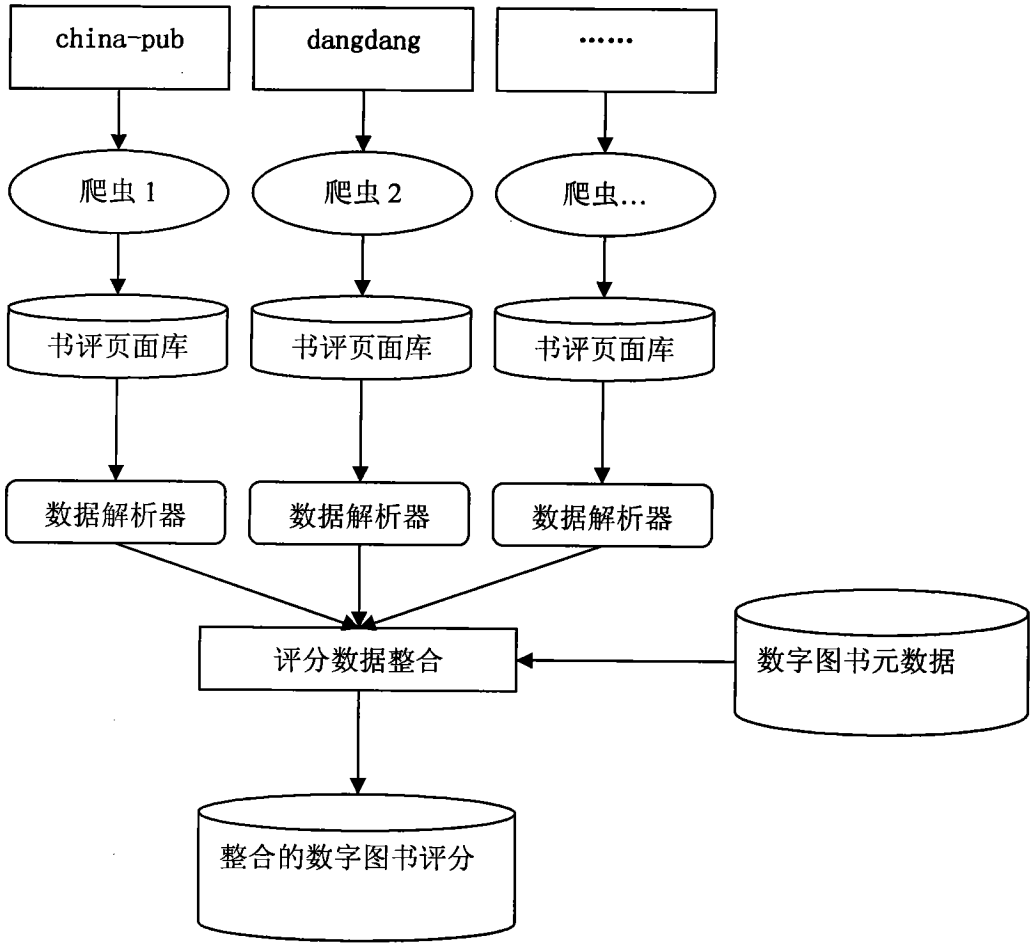

china-pub
dangdang
……
爬虫 1
爬虫 2
爬虫…
书评页面库
书评页面库
书评页面库
数据解析器
数据解析器
数据解析器
评分数据整合
数字图书元数据
整合的数字图书评分

图 1