

一种基于动态用户模型的个性化 Web 搜索算法

袁晓洁 窦志成 刘芳 张路

(南开大学信息技术科学学院, 天津 300071)

(yuanxj@nankai.edu.cn)

摘要 个性化 Web 搜索能够根据用户兴趣偏好为不同用户返回更符合其信息需求的个性化查询结果, 是解决查询歧义性问题, 提高搜索精度和用户满意度的有效途径。但用户查询复杂多样, 基于用户所有历史查询而建立的静态用户模型往往不准确, 降低了个性化搜索的性能。针对这一问题, 提出了一种基于动态用户模型的个性化算法。在建立该用户模型时, 将与当前查询相似性较高的历史查询赋予较高的权重, 去除不相关查询, 提高用户模型的可靠性和准确性。实验结果表明基于动态用户模型的个性化算法性能要明显优于基于静态用户模型的算法, 并且能够有效地提高搜索精度。

关键词 个性化 Web 搜索; 用户模型; 用户兴趣; 搜索引擎

中图法分类号 TP393

Personalized Web Search Based on Dynamic User Profile

Yuan Xiaojie, Dou Zhicheng, Liu Fang, and Zhang Lu

(College of Information Technical Science, Nankai University, Tianjin 300071)

Abstract Personalized Web search is an effective way to solve the problem of query ambiguity because it is able to return different search results to different users based on their preferences. Because user search history involves much noisy information that is irrelevant to user interests, static user profile built upon all historical user queries is usually inaccurate and may harm search performance sometimes. To solve this problem, this paper proposes a personalized Web search algorithm based on a dynamic user profile. When generating this user profile, historical queries that are related to the current query are given high weights and other unrelated queries are ignored. Experimental results show that personalized search algorithm which uses dynamic user profile outperforms the one which uses static user profile, and could improve search performance significantly.

Keywords personalized web search; user model; user interest; search engine

个性化搜索是解决查询歧义性问题, 提高搜索精度和用户满意度的有效途径。目前的搜索引擎系统大都没有考虑用户的差异。对于所有用户, 只要输入的查询词相同, 则返回的结果就完全相同。而事实上, 很多查询都存在歧义性。由于背景知识、兴趣爱好等方面的差异, 不同的用户在使用同一查询时, 也往往是在检索不同的信息。例如, 考古学家使用查询词“甲骨文”来查询中国古文字相关信息, 而数据库系统开发人员则可能用该查询来查找甲骨文公司(Oracle)或

甲骨文数据库的相关信息。表 1 显示了在搜索引擎 Google.com 中查询“甲骨文”时返回的前 6 个结果。可以看到, 关于中国古文字和甲骨文公司的信息是掺杂在一起的, 用户仍然需要在返回的结果集中挑选出真正感兴趣的内容。个性化搜索能够根据用户的知识背景、兴趣爱好、浏览习惯等信息, 估计用户查询目的, 对查询结果进行重新排序和过滤, 从而为不同的用户返回更符合其信息需求的查询结果集, 提高搜索精度和用户满意度。

收稿日期: 2008-05-25

基金项目: 天津市科技发展计划基金项目(06YFGZGX05700); 天津市应用基础研究计划项目(07JCYBJC14500)

表1 Google.com 中查询“甲骨文”的前6个结果

序号	网页标题	类别
1	甲骨文中国公司	公司
2	Oracle 公司	公司
3	计世网--企业品牌中心--Oracle	公司
4	甲骨文_明星公司_财经纵横_新浪网	公司
5	殷墟 甲骨文 安陽市 人民政府	文字
6	甲骨文与金文	文字

现有个性化搜索算法主要通过建立用户兴趣模型(User Profile, User Model), 比较用户模型与搜索结果的相似性, 对搜索结果进行个性化处理^[1-6]。因为用户一般不愿意主动提供和更新个人信息, 大部分现有个性化搜索算法都基于用户查询和点击历史自动学习和更新用户兴趣模型^{[4][5][8]}。基于用户兴趣的个性化算法能够为用户过滤出符合其兴趣爱好的文档, 在一定程度上提高了搜索精度和用户体验。如对于查询“甲骨文”, 针对软件开发人员, 仅返回关于甲骨文数据库公司的文档, 而对于一个古文字研究者, 仅返回关于中国古文字的信息。但这种算法在实际应用中却存在着问题——用户查询信息往往是因为环境影响而产生的临时需要, 并不总是和其兴趣爱好一致的。如用户生病时可能会查询关于疾病的信息, 因为计算机感染病毒而查找关于电脑病毒的信息。用户还有可能代替别人查询信息。在这些情况下产生的用户查询和用户的兴趣爱好并没有任何关系。首先, 如果基于用户的兴趣对这些查询进行个性化, 得到的结果很可能是错误的。其次, 如果将这些信息融入到用户模型中, 会降低用户模型的准确性, 对将来的用户查询产生影响。而大部分已有研究工作都没有考虑到这一问题, 只是简单的基于所有用户历史查询建立用户模型(本文称这种用户模型为**静态用户模型**)。文献^[12]在大规模真实搜索引擎上的实验表明, 这种基于静态用户模型的个性化搜索算法在实际搜索引擎中不仅不能提高反而会降低搜索精度。

针对上述问题, 本文提出了一种动态用户兴趣模型。在该用户模型中, 并不是简单地使用所有用户历史查询, 而是将和当前查询相关的历史查询赋予较大的权重, 而忽略和当前查询无关的历史查询。因为该用户兴趣模型是和当前用户查询相关的, 需要在用户查询时动态建立, 因此称该用户模型为动态用户模型。本文基于领域主题分类分别建立了静态用户兴趣模型和动态用户模型。在大规模真实用户查询上的实验表明, 基于动态用户兴趣模型的个性化算法要明显优于基于静态用户兴趣模型的算法, 并且能够有效提

高用户查询的排序精度。

本文第1节描述静态用户兴趣模型和动态用户模型。第2节介绍了一种基于该用户模型的个性化搜索算法。第3节和第4节描述实验数据、实验过程并分析实验结果。第5节总结全文并进行展望。

1. 用户兴趣模型

用户兴趣模型一般采用用户描述文件(user profile)表示。根据算法的需要, 用户描述文件可以组织为不同的内容和格式。目前典型的用户描述文件内容包括关键词列表^{[5][7][8]}、主题分类列表^{[1][2][3][4]}和偏好网页列表^[13]等。本文提出了一种基于主题分类向量的用户描述文件格式。用户兴趣被分类到多个主题中。每个主题具有对应的权重来表示该主题的相关程度。文档也同样被分类到不同的主题中。查询时, 通过比较文档主题分类和用户兴趣的相似性, 判断文档对用户是否相关。使用该类型的用户描述文件能够识别所有符合主题的文档, 预测能力较强。

1.1 文档主题分类向量

假设 Web 上的文档被分为 R 个不同的主题。采用 \mathbf{d}_n 表示文档 n 的 R 维主题分类向量(注, 本文使用黑体表示矩阵或向量), 有:

$$\mathbf{d}_n = [d_{n,1}, d_{n,2}, \dots, d_{n,R}]^T \quad (1)$$

其中, $d_{n,r}=f$ 代表文档 n 属于主题 r 的可能性为 $f(0 \leq f \leq 1)$, 且 $|\mathbf{d}_n| = \sum_{j=0}^R d_{n,j} = 1$, 即 \mathbf{d}_n 是一个概率向量。

本文采用了文献^[9]提供的领域主题分类层次结构中的第二层主题类别。文档被分为 67 个不同的主题, 即 $R=67$ 。本文使用了文献^[10]中提出的查询和文档分类算法, 开发了文本分类器。给定一个文档, 该分类器返回前 6 个可能属于的分类, 并对应给出属于每个分类的可信度。按如下公式计算 $d_{n,j}$:

$$d_{n,j} = d_{n,j}^o + \frac{1}{R} \left(1 - \sum_{k=1}^R d_{n,k}^o \right) \quad (2)$$

其中 $d_{n,j}^o$ 代表分类器返回的文档 n 分类到主题 j 的可信度(概率)。如果主题 j 不在分类器返回的 6 个主题中, 则 $d_{n,j}^o=0$ 。

1.2 静态用户兴趣模型

本节介绍静态用户兴趣模型。为简单起见, 下文所用符号中均省略用户符号。假设用户共提交过 M 个不同的查询, 点击过 N 个不同的文档。使用 M 维向量 \mathbf{Q} 代表用户的查询向量, 使用矩阵 $\mathbf{C}_{M \times N}$ 表示用户查询的文档点击分布。其中 \mathbf{Q} 中元素 $Q_m=x$ 表示用户一共提交过 x 次查询 m 。矩阵元素 $C_{m,n}=y$ 代表用户

在查询 m 上一共点击过 y 次文档 n 。

鉴于不同用户认为同一个查询所属的分类有可能是不同的, 本文不采用文本分类的方法直接对查询进行分类, 而是基于查询中用户点击的文档的主题分类来计算用户查询的主题分类。用户查询 m 的主题分类向量 q_m 按如下公式生成:

$$q_m = \sum_{i=0}^n \frac{C_{m,i}}{\sum_{j=1}^N C_{m,j}} d_i = \frac{\sum_{i=1}^n C_{m,i} d_i}{\sum_{j=1}^N C_{m,j}} \quad (3)$$

其中 $\frac{C_{m,i}}{\sum_{j=1}^N C_{m,j}}$ 代表在查询 m 中用户点击文档 i 的概率。易知 q_m 也是一个概率向量。

接下来, 基于用户查询的主题分类计算用户的兴趣模型。使用 U 代表用户的兴趣向量, 有:

$$U = \sum_{m=1}^M w_m q_m \quad (4)$$

其中 w_m 是在计算用户描述文件时查询 m 采用的权重, 且:

$$w_m = \frac{Q_m}{\sum_{i=1}^M Q_i} \quad (5)$$

和文档 n 的主题向量 d_n 类似, U 是一个 R 维概率向量。该用户模型仅和用户的历史查询与点击文档相关, 而和当前用户查询无关, 因此称之为静态用户兴趣模型。

1.3 动态用户兴趣模型

在第 1.2 节中介绍的用户兴趣模型中, 假设每个查询对用户兴趣的影响是相同的。但事实上, 如第 1 节所述, 并不是每个查询都和当前用户查询是相关的。用户查询一般是多种多样的, 简单的使用所有历史查询对当前用户查询进行个性化有时不仅不能改善搜索精度, 甚至可能会扰乱搜索结果, 而只有那些和当前查询有联系的查询才是有效的。例如, 如果用户曾经查询过“奥运会”和“Web 搜索”, 而用户当前查询是“信息检索”。很显然查询“奥运会”和“信息检索”是不相关的, 如果在建立用户模型时使用查询“奥运会”的主题分类向量, 不仅不能改善查询“信息检索”的排序精度, 反而影响历史查询“Web 搜索”在用户模型中的作用, 降低个性化算法的性能。

针对上述问题, 本文提出了一种新的用户兴趣模型。在该模型中, 将和当前查询相关的历史查询赋予较大的权重, 而忽略和当前查询无关的历史查询。因为该用户兴趣模型是和当前用户查询相关的, 因此称之为**动态用户模型**。

对公式(4)进行改进, 在建立用户模型时, 为每个用户历史查询加入一个影响因子 λ 。

$$U(q) = \sum_{m=1}^M \lambda_m w_m q_m \quad (6)$$

其中 λ_m 可以看作是历史查询 m 和当前用户查询 q 的相似性。因为查询词的长度一般较短, 直接根据查询词直接计算查询之间的相似性的准确性较差, 而且不能解决同义词和近义词等问题(如查询 oracle 和甲骨文)。为解决这一问题, 本文采用查询的搜索结果摘要(文档标题和显示在搜索结果列表中的文档与查询相匹配的文档片段)代替查询词本身。本文之所以采用文档摘要而并未采用文档本身, 是因为发现文档摘要更能够体现出当前查询要表述的内容, 包含的噪声较小。

对每个查询 q , 设其搜索结果摘要的集合为 $L(q)$ 。在实际应用中, 可以根据需要只选取前 n 个搜索结果。将 $L(q)$ 中的所有文本拼接在一起生成一个大文本串作为查询 q 的虚拟文档。对该文档建立对应的 TF-IDF 向量, 称之为查询向量, 并用 $v(q)$ 表示。计算查询之间的相似性时, 使用查询向量来代替原始查询文本。

采用余弦相似性计算两个查询向量的相似程度:

$$\lambda_m = \cos(q, m) = \frac{v(q) \cdot v(m)}{\|v(q)\| \|v(m)\|} \quad (7)$$

将公式(7)带入到公式(6)中, 计算出与当前查询相关的动态用户模型 $U(q)$:

$$U(q) = \sum_{m=1}^M \cos(q, q_m) w_m q_m = \sum_{m=1}^M \frac{v(q) \cdot v(m)}{\|v(q)\| \|v(m)\|} \frac{Q_m}{\sum_{i=1}^M Q_i} q_m \quad (8)$$

2. 基于动态用户模型的个性化算法

第 1 节分别介绍了文档模型和用户模型。在查询时, 通过计算文档模型和用户模型的相似性对结果文档进行重排, 生成个性化的结果排序。本文使用向量的余弦相似度来计算文档与用户模型的相似性。具体的, 静态用户模型 U 与文档模型的相似性计算按下面公式计算:

$$Sim(U, d_n) = \cos(U, d_n) = \frac{U \cdot d_n}{\|U\| \cdot \|d_n\|} \quad (9)$$

类似的, 动态用户模型 $U(q)$ 和文档模型的相似性按如下公式为:

$$Sim(U(q), d_n) = \cos(U(q), d_n) = \frac{U(q) \cdot d_n}{\|U(q)\| \cdot \|d_n\|} \quad (10)$$

具体的个性化搜索流程如算法 1 所示。为方便起见, 本文对基于静态用户模型与动态用户模型的方法

在算法1中统一进行了介绍:

算法1. 基于静态和动态用户模型的个性化搜索算法

输入: 文档主题分类模型, 用户模型 (使用静态用户模型时需要), 用户查询, 一个搜索引擎。

输出: 个性化搜索的结果排序列表

算法步骤:

- ① 接受用户查询, 使用搜索引擎产生初步搜索结果集 R 。使用 τ_{ori} 代表搜索引擎返回的结果排序列表。
- ② 置迭代次数 $i=0$ 。
- ③ 对集合 R 中的第 i 篇文档, 利用公式(2)计算其的文档分类矢量 d_i 。
- ④ 使用公式(8)计算动态用户模型 $U(q)$ (使用静态用户模型时忽略此步)。
- ⑤ 利用公式(9)或(10)计算静态用户模型 U 或动态用户模型 $U(q)$ 与文档 i 的相似性。
- ⑥ 如果文档 i 是集合 R 中最后一篇文章, 转⑦; 否则, 置 $i=i+1$, 返回③。
- ⑦ 使用计算出的相似性对 R 中的文档进行排序, 生成个性化排序列表 τ_{per} 。
- ⑧ 使用 Borda 排名组合方法^[10]对 τ_{ori} 和 τ_{per} 进行组合, 并生成最终排序列表 τ_{rerank} 。 τ_{rerank} 即为最终的结果排序列表。
- ⑨ 将 τ_{rerank} 返回给用户。

该算法的第⑧步将个性化结果排序和初始搜索结果排序进行了组合。之所以引入初始结果排序, 是为了提高个性化算法的稳定性。在实验中, 本文仅选取 R 为全部搜索结果的前 100 个文档的集合。这是因为在实际搜索引擎中用户极少点击排名在 100 以后的文档, 这也是个性化搜索算法常用的方式。在实际应用中, 文档主题分类向量一般可提前算出, 部分查询向量如查询频率较高的查询也可以提高计算并缓存以提高算法效率。同时因为查询向量是基于搜索结果的, 在实际应用到搜索引擎中时, 可结合到检索模型中, 根据倒排表等结构高效实时计算。

3. 实验方法与性能评估

3.1 实验方法与实验数据

和文献[12]的实验方法类似, 本文采用搜索点击日志中的用户点击作为评价标准对个性化搜索算法性能进行评价。该评价方法中, 近似认为用户点击的文档是符合用户查询需求的文档。本文采用某大规模搜索引擎 12 天的查询日志。该查询日志中记录了用户提交的查询与点击的文档 URL, 其中用户采用 Cookie 进行标识。首先从中随机选取了 10,000 个用户, 然后将这些用户的所有查询和点击日志抽取出来

构成本实验的数据集。然后将该数据集分为两部分, 前 11 天为训练集, 作为用户查询历史建立用户模型; 最后一天为测试集, 对该天的查询应用个性化算法并使用用户点击进行评价。最终, 该数据集中包含查询 89,865 个, 其中训练集中包含 82,755 个查询, 测试集中包含 7,110 个查询。去掉其中没有任何点击的查询和在搜索引擎中无法找到对应搜索结果的查询后, 共剩余 3,963 个测试查询。在这些查询中, 约 57% (2,256/3,963)的查询上用户点击的文档都处于结果列表顶部, 也就是说, 本文使用的搜索引擎在这些查询上已经做到了最优, 个性化算法已经不能再对这些查询进行任何改进。称这些查询为最优查询, 称其他 1,707 个查询为非最优查询。

本文采用现有搜索引擎作为个性化算法性能比较的基准方法。

3.2 性能评估指标

本文采用文献[12]和文献[13]中使用的排序计分指标(Rank Scoring)对算法性能进行评价。一个文档列表排序的期望效益定义为:

$$R_s = \sum_j \frac{\delta(s, j)}{2^{(j-1)/(\alpha-1)}} \quad (11)$$

其中 j 是文档在结果列表中的排序号。如果用户在查询 s 中, 点击了文档 j , 则 $\delta(s, j)$ 为 1, 否则为 0。和文献[16]相同, 本文取 $\alpha=5$ 。所有测试集中用户查询的整体效益为:

$$R.S. = 100 \frac{\sum_s R_s}{\sum_s R_s^{Max}} \quad (12)$$

其中 R_s^{Max} 是最大可能期望, 当所有用户点击过的文档都出现在列表顶端时获得。 $R.S.$ 称为排序计分 (Rank Scoring)。

4. 实验结果与性能分析

为了能够分析基于动态用户模型的个性化算法的性能, 本文将其与基于静态用户模型的算法进行了比较。对测试集中的查询分别采用基于静态用户模型的算法和基于动态用户模型的算法进行了个性化处理。表 2 给出了实验结果。该表说明在所有查询、非最优查询及最优查询上, 使用静态用户模型的方法都显著地降低了搜索性能。这说明在实际搜索引擎中, 用户查询是复杂多样的, 基于静态用户模型的方法非常不稳定, 不但不能有效改进搜索精度, 还会明显地降低搜索精度。和基于静态用户模型的方法相比, 基于动态用户模型的方法则稳定的多。在所有查询上对搜索精度稍有提高 (约 2.06%, 通过 T 检验 $p<0.05$), 在非最优查询上的改进更为明显 (约 6.69%), 而在最优查询上对搜索精度的降低也较小。

表 2 静态用户模型和动态用户模型性能比较

类型	测试查询 数目	基准 方法	静态 用户模型		动态 用户模型	
所有查询	3,963	69.47	52.00	-25.15%	70.90	2.06%
非最优查询	1,707	47.26	37.08	-21.54%	50.42	6.69%
最优查询	2,256	100.00	72.50	-27.50%	99.44	-0.56%

文献[12]中指出, 个性化算法在歧义性较高的查询上性能较好, 而在歧义性较低的查询上使用时作用较小。点击熵可以作为区别歧义性查询和非歧义性查询的简单标准。表 3 中给出了算法在点击熵小于 1.5 和大于等于 1.5 的查询上的性能比较。从该表中可以看出, 基于静态用户模型的个性化算法在点击熵小于 1.5 的查询上明显地降低了搜索精度; 在点击熵大于 1.5 的查询上虽然能够改进搜索精度, 但改进非常有限。而基于动态模型的个性化算法的性能要明显优于基于静态用户模型的算法: 在点击熵小于 1.5 的查询上, 算法性能相比原始搜索算法性能并没有下降, 反而略有改进; 在点击熵大于 1.5 的查询上, 搜索精度比基准搜索方法提高了 12.93% (T 检验 $p < 0.05$), 这说明它能够有效地改进歧义性查询的搜索精度。

表 3 静态用户模型和动态用户模型性能比较

点击熵	基准方法	静态用户模型		动态用户模型	
<1.5	70.4776	52.3941	-0.47%	71.789	1.86%
>=1.5	43.6514	43.8552	0.17%	49.2946	12.93%

5. 总结与展望

个性化搜索是解决查询歧义性问题, 提高用户搜索精度的有效方法, 是搜索引擎技术发展的趋势。本文指出了在用户查询中存在大量噪声, 基于所有用户历史查询建立的静态用户模型不准确, 不能有效改进搜索精度的问题, 并针对这一问题提出了一种动态用户模型。首先根据查询的搜索结果摘要建立查询的 TF-IDF 向量, 通过向量余弦相似性计算用户历史查询与当前查询的相似性。在建立动态用户模型时, 按照用户历史查询与当前查询的相似性高低为不同的历史查询赋予不同的权重。本文分别实现了基于主题向量的静态用户模型和动态用户模型, 并基于大规模点击日志对它们的性能进行了对比。实验结果表明基于动态用户模型的个性化算法明显优于基于静态用户模型的方法, 并且能够有效地改进高歧义性查询上的搜索精度。在点击熵大于 1.5 的查询上, 基于动态用户模型的方法和不用个性化的普通搜索方法相比, 搜索精度提高了约 13%。

参考文献

- [1] A. Pretschner and S. Gauch. Ontology based personalized search. In Proceedings of ICTAI '99, pages 391–398, 1999.
 - [2] Zeng C, Xing CX, Zhou LZ. A Personalized Search Algorithm by Using Content-Based Filtering. Journal of Software, 2003,14(05):999~1004 (in Chinese with English abstract).
 - [3] P. A. Chirita, W. Nejdl, R. Paiu, and C. Kohlschütter. Using ODP metadata to personalize search. In Proceedings of SIGIR '05, pages 178–185, 2005.
 - [4] M. Speretta and S. Gauch. Personalized search based on user search histories. In Proceedings of WI '05, pages 622–628, 2005.
 - [5] B. Tan, X. Shen, and C. Zhai. Mining long-term search history to improve search accuracy. In Proceedings of KDD '06, pages 718–723, 2006.
 - [6] Zeng C, Xing CX, Zhou LZ. A survey of personalization technology. Journal of Software, 2002,13(10):1952~1961 (in Chinese with English abstract).
 - [7] X. Shen, B. Tan, and C. Zhai. Context-sensitive information retrieval using implicit feedback. In Proceedings of SIGIR '05, pages 43–50, 2005.
 - [8] X. Shen, B. Tan, and C. Zhai. Implicit user modeling for personalized search. In Proceedings of CIKM '05, pages 824–831, 2005.
 - [9] Y. Li, Z. Zheng, and H. K. Dai. Kdd cup-2005 report: facing a great challenge. SIGKDD Explor. Newsl.,7(2):91–99, 2005.
 - [10] D. Shen, R. Pan, J.-T. Sun, J. J. Pan, K. Wu, J. Yin, and Q. Yang. Q2c@ust: our winning solution to query classification in kddcup 2005. SIGKDD Explor. Newsl., 7(2):100–110, 2005.
 - [11] C. Dwork, R. Kumar, M. Naor, and D. Sivakumar. Rank aggregation methods for the web. In Proceedings of WWW '01, pages 613–622, 2001.
 - [12] Zhicheng Dou, Ruihua Song and Ji-Rong Wen, A Large-scale Evaluation and Analysis of Personalized Search Strategies, In Proceedings of WWW '07, pages 581–590, 2007.
 - [13] J.-T. Sun, H.-J. Zeng, H. Liu, Y. Lu, and Z. Chen. CubeSVD: a novel approach to personalized web search. In Proceedings of WWW '05, pages 382–390, 2005.
 - [14] Salton, G., Fox, E. A. and Wu, H. 1983 Extended Boolean information retrieval. Commun. ACM 26, 1022–1036
 - [15] Salton, G. and Buckley, C. 1988 Term-weighting approaches in automatic text retrieval. Information Processing & Management 24(5): 513–523.
- 袁晓洁**, 女, 1963 年生, 教授, 博士生导师, 研究领域: 数据库, 数据挖掘, 信息检索。
- 窦志成**, 男, 1980 年生, 博士研究生, 研究方向: 信息检索, 数据挖掘, 个性化 Web 搜索, 日志分析。
- 刘芳**, 女, 1983 年生, 硕士研究生, 研究方向: 分布式存储。
- 张路**, 男, 1983 年生, 博士研究生, 研究方向: 分布式存储。
- 负责人联系方式: 南开大学, 窦志成, 022-23508468/13601258246, 天津市南开大学信息技术科学学院 zhichengdou@yahoo.com.cn