网络爬虫性能研究*

漆志辉,杨天奇

(暨南大学 信息科学技术学院 计算机系,广东 广州 510632)

摘 要:受到学习模型爬虫的启发,主题爬虫结合网页内容和链接信息来估计网页对给定主题的相关性,得到两个新型的爬虫变种。新型爬虫强调的不仅是有学习相关网页内容的能力,而且有引向相关网页的能力,并且在查找特定主题方面的能力有质的提高。

关键词:主题爬虫;学习型爬虫;学习型主题爬虫

中图分类号:TP301 **文献标识码**:A **文章编号**:1674-7720(2011)05-0072-03

Research of the function of network crawlers

Qi Zhihui, Yang Tianqi

(Department of Computer, Institute of Information Science and Technology, Jinan University, Guangzhou 510632, China)

Abstract: Inspired by learning crawler, this paper obtains two new focused crawlers which combine Web page content and link information. The new focused crawlers emphasis not only on the capability of learning the content of relevant pages but also paths leading to relevant pages. Furthermore, the new crawlers' ability to find more specific topics has improved.

Key words: focused crawler; learning crawler; learning focused crawler

随着因特网技术的发展,传统的通用搜索爬虫正面临着巨大的挑战,已经不能满足人们对个性化信息检索服务日益增长的需要。专业搜索引擎搜索的内容只限于特定主题或专门领域,因而在搜索过程中无须对整个Web进行遍历,只需选择与主题页面相关的页面进行访问。

主题爬虫的搜索策略常见的有5种:(1)基于内容评价的搜索策略。这类网络蜘蛛在距离相关页面集较近的地方搜索时表现出良好的性能。但由于页面中的文本信息缺乏"全局性",很难反映Web的整体情况,普遍存在"近视"的缺点。(2)基于链接结构评价的搜索策略。这种策略利用页面之间的引用关系确定链接的重要性。这类搜索策略优点是考虑了链接的结构特征,缺点是忽略了页面与主题的相关性,在某些情况下会出现搜索偏离主题的"主题漂移"问题。此外,其在搜索过程中需要重复计算PageRank值或Authority及Hub权重,计算复杂度随访问的页面和链接数量的增长呈指数级增长。(3)基于未来回报价值评价的搜索策略。这种策略本质上是通过训练发掘出链接文本中"隐含"的结构信息,这些结构信息反映了距离搜索目标的远近,因而在搜索远期回报方面具有一定优势。然而,这类搜索策略也存在一些不足:一是预测未来回报能力有限;二是这种"离线"的训练方式需要选择典型站点或种子集,加重了用户的负担。(4)基于"综合价值"评价的搜索策略。采用单一的评价方法不能有效预测链接的真实价值。这类搜索可以有效提高搜索效率。(5)基于动态价值评价的搜索策略。根据环境的变化动态调整价值评价机制,表现出极大的灵活性。

根据搜索策略的不同可以把主题爬虫归为下面几类:

(1)传统主题爬虫[1]将描述主题的用户查询语句作为其输入,这是一些种子网页URL集,并且它会把查找导向感兴趣的网页。这种爬虫的文本相似度是用信息相似度模型来计算的,这些模型有布尔型模型和向量空间模型(VSM)[2]。

(2)语义型爬虫[3]是传统主题爬虫的变种。根据语义相似度标准,把下载权重分配给页面,这样就可以计算出页面内容和主题的相关度:如果页面和主题都有概念上(没必要是词语上的)相似的短语,那页面和主题具有

*基金项目:广东省软科学研究项目(2009B070300052)

相关性。短语之间的概念相似度是使用本体论[4]来定义的。

(3)学习型爬虫[5]采用训练过程来给网页指派访问权重和引导抓取过程。这类爬虫的特点是爬虫学习了网页相关方式或者通过网页链接来到达相关页面的路径。

将学习型爬虫的思想和传统主题爬虫的思想进行合理结合,这样改造出来的新型爬虫就同时具有学习型爬虫和传统主题爬虫的优点。受到 HMM 爬虫的启发,学习型爬虫结合采用网页内容和链接信息来估计网页对给定主题的相关性,这样就可以得到新型的爬虫变种。

1 爬虫设计与实现

爬虫的设计与实现:(1)输入。爬虫的输入包括一定数量的初始种子 URL 和主题描述词。主题描述词可以是关键词的列表。(2)下载网页。抽取网页中的活跃链接,并将其置于队列中。主题爬虫的队列排序和传统爬虫不一样,需要根据一定的标准重新排序。(3)处理网页内容。对网页进行分词处理,分解成词语向量,采用向量空间模型(VSM)来计算文本相似度。(4)权重分配。从网页中抽取到的活跃链接放在一个权重队列中,权重队列中的权重分配是由爬虫的类型和用户的喜好决定的。(5)重复步骤(1)~(4)。选择 URL 进行进一步的爬行,重复步骤(1)~(4)直到满足一些停止爬行的条件,或者系统资源耗尽。

HMM 爬虫[6]的工作是建立网页内容与导向相关页面路径之间的关系。首先用户浏览一个特定的主题页面,并且对网页进行标记相关或者不相关,保存这些页面以建立页面训练种子集。相关页面组成簇(D_0)。不相关的页面采用 K-Means[7](K 由用户定义)分簇,它们形成簇 D_1~D_k。HMM 模型建立的分簇基础是:每个页面有两个状态特征:(1)显状态。根据网页的内容来确定页面属于哪个簇;(2)隐状态。页面和目标页面的距离。假定页面属于这个簇,那这个簇的权重是它能导向目标页面的概率。

图 1 中展现了 HMM 爬虫训练集。L_0 表示目标或 0 级网页,L_1 是 1 级页面(与目标页面相距 1 个链接),L_2 是 2 级页面(与目标页面相距 2 个链接),L_3 是与目标页面相距 3 个或更多链接。D_0、D_1 和 D_2 标签分别对应簇 0、1、2。有相同簇的页面可能属于不同页面级,在同一页面级的页面可能属于不同的簇。

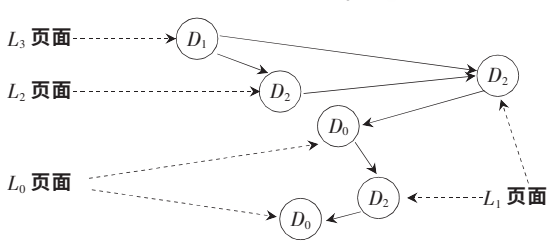

图 1 HMM 训练集

HMM 爬虫用到的参数和记号:网页的等级或隐状态特征 $Li(i$ 是等级),显状态用它们归属的簇 D_j 来表示。页面集隐状态和显状态可以用 HMM 模型来建模。

(1)初始概率矩阵:$\Phi=\{P(L_0,\cdots,L_{s-1})\}$,其中 s 是隐状态的数目,$P(L_i)$表示在时间 1,处于隐状态 i 的概率。这个概率的计算方法是:分配给每个页面一个值,这个值等于在训练集中有相同隐状态的页面的比例。

(2)过渡概率矩阵:$\boldsymbol{A}=[p_{ij}]_{0\leqslant i<s,0\leqslant ij<s}$,如果在状态 L_i 时间 t,p_{ij} 则表示在状态 L_j 时间 $t+1$ 时的概率。这个概率假定和 t 独立,且通过计算从在训练集中状态 L_i 到 L_j 的过渡矩阵个数和通过把来自状态 L_i 的全部过渡矩阵归一化,即可以得到这个概率。

(3)发射概率矩阵:$B=[b_{ij}]_{0\leqslant i<s,0\leqslant j<c}$,其中 b_{ij} 代表指定状态 L_i 在簇 D_j 中的概率,c 是簇的个数。这个概率的计算方法是:先计算在簇中有隐状态 L_i 的个数,然后把在隐状态 L_i 的网页总数归一化。

在基于训练集的抓取开始之前,Φ、$\boldsymbol{A}$、B 已经计算出来了。爬虫下载页面后,抽取它们的向量表示(根据 VSM),并且用 K-Nearest Neighbors 算法给每个网页分簇。页面的簇确定后,这个页面能抓取到目标页面的概率可以用 HMM 模型参数来计算。这个概率与包含在那个页面中的链接访问权重一致。若要考虑页面的次序,采用 HMM 模型来预测在下一个时间步骤里的状态。为了计算预测值,每个访问过的页面与值 $p(L_j,t),j=0,1,\cdots,s$ 相关联。值 $p(L_j,t)$是爬虫在时间 t 时刻下载具有隐状态 L_j 页面的概率。指定父页面值 $p(L_j,t-1)$,值 $p(L_j,t)$ 采用下面递归公式计算:

$$p(L_j,t)=b_{jct}\sum_{i=0}^{s}(p(L_i,t-1)\cdot p_{ij}) \tag{1}$$

其中,p_{ij} 是在矩阵 $\boldsymbol{A}$ 中从状态 L_i 到 L_j 的过渡概率,b_{jct} 是来自矩阵 $\boldsymbol{B}$ 中隐状态的簇 c_t 的发射概率。在最后递归步骤中,值 $p(L_j,0)$取自初始化概率矩阵 Φ。指定值 $p(L_j,t)$,即在下一个时间步所选页面将会处于状态 L_j 的概率的计算公式:

$$p(L_j,t+1)=\sum_{i=0}^{s}(p(L_i,t)\cdot p_{ij}) \tag{2}$$

在下一个步骤处于状态 L_0 的概率是 HMM 爬虫分配给网页的权重。如果两个簇产生相同的概率(例如它们的概率差值低于预定义的阈值 ε),那么更高的权重分配给那些具有可以在两步内(同样用式(1)和(2)计算)导向目标页面概率更高的簇。在导向它们的路径中,与相同簇次序相关联的页面分配相同的权重。改进后的爬虫,页面权重分数规定为用 HMM 爬虫和计算的权重及代表页面的由短语向量表示相关的分类(质心)向量的相似度的平均数。新型的 HMM 爬虫变种只采用页面内容,或同时采用页面内容和链接文本。

图 2 展示了主题爬虫的操作步骤。页面 P_1(簇 C_1 中)

和 P_2(簇 D_2 中)是下载的候选页面。簇 D_1 和 D_2 都有在一步内导向簇 D_0(目标页面)相等权重。然而,这两个簇都可能在两个链接步内到达目标页面。原始的爬虫会分配更高的权重给页面 P_1(页面 P_2 可能在两个链接步内也到达 D_0,与在 D_3 中非目标页面一样,分配给页面 P_2 低权重)。拟议的爬虫(图 2)宁愿选择 P_2,因为它接近簇 D_0(直观上更正确)的质心 D_r。下面总结了新型爬虫的操作步骤:(1)输入:训练集,候选页(P);(2)输出:权重值分配给候选页面 P;(3)采用 K-Means 训练集的簇(K 由用户定义);(4)计算 Φ,A,B 矩阵和相关页面的质心 D_r;(5)采用 K-Nearest Neighbor 算法,将候选页面 P 分类到簇 D_t;(6)给当前步骤计算隐状态概率:$p(L_j,t)=b_{jct}\sum_{i=0}^{S}(p(L_i,t-1)\cdot p_{ij})$;(7)计算下一步隐状态概率估计:$p(L_j,t+1)=\sum_{i=0}^{S}(p(L_i,t)\cdot p_{ij})$;(8)计算权重;(9)采用 VSM 计算页面内容(或者页面内容和链接文本)和相关页面的质心 D_r 之间的关系;(10)分配权重给网页:$pri_{learningIMM}(p,D_r)=(sim(p,D_r)+pri_{IMM}(P))/2$。

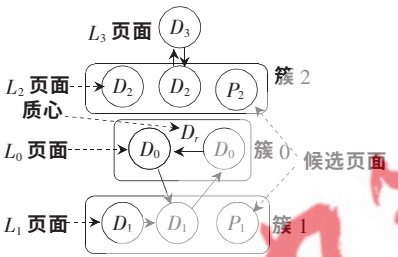

图 2 新型爬虫的操作步骤

2 实验结果

2.1 实验设置

所有的爬虫都用 C++实现。要下载的页面必须是 text/html 格式,其内容大小不超过 300 KB。由于性能的因素,链接超时和下载时间同样也要考虑。所有已实现的爬虫都有这些限制。抓取过程一直重复,当抽取到预定页面数量(1 000)时,则结束。实现且评估前面提到的所有爬虫,让它们抓取的主题相同。

爬虫的性能,由下载到的页面中和主题相关的页面比例决定(如相似度大于预定的阈值的页面,本文中阈值取 0.75)。这项措施称为“收获率”。收获率可以用来调整测量爬虫下载和主题高度相关页面的能力。

初始的种子页面由人工完成。把相关的页面组成主题的种子页面,每个主题的种子页面集大小为 100。对于每个主题,把爬虫抓取到的结果和种子页面作比较,因为对爬虫返回的每个页面,采用 VSM 方法计算它们的文档相似度,如果它们的相似度值的最大值比用户定义的阈值要大,那么这个页面就标记为正结果。爬虫的正结果越多,这个爬虫就越成功,即爬虫抓取到和主题相似的结果的概率就更高。爬虫的性能是所有主题的正结果数的平均数。

2.2 爬虫评估

本文对以下三种爬虫进行了评估:(1) 原始的 HMM 爬虫;(2)HMM 爬虫采用页面内容相似度,相似度具有相关页面簇质心;(3)HMM 采用页面内容和链接文本相似度,相似度具有相关页面簇质心。

三种爬虫的结果比较如图 3 所示。

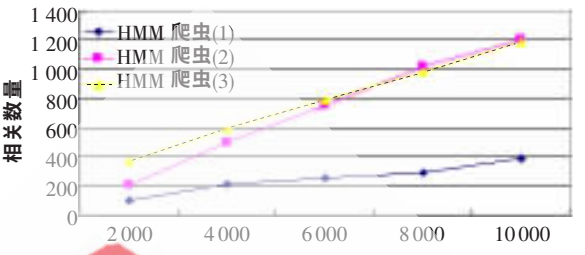

图 3 三种爬虫结果比较

从图 3 可以看到,改进后爬虫的所有实现胜过传统的 HMM 爬虫,当允许它们根据页面的内容分配给页面不同的优先权时,这些页面在导向它们的路径中有相同的簇次序(即使在一个页面中的链接,在使用链接文本的时候)。表 1 是三种爬虫的平均运行时间和页面相关率统计表。

表 1 运行时间和相关率统计表

爬虫种类	平均运行时间/min	页面相关率/%
HMM 爬虫(1)	1 506.7	4.11
HMM 爬虫(2)	1 660.5	13.34
HMM 爬虫(3)	1 732.6	13.17

从表 1 可以看到,传统的爬虫运行时间是最短的,但它抓取到的网页页面相关率只有 4.11%。两种改进后的爬虫——HMM 爬虫(2)和 HMM 爬虫(3),其运行的时间相对较长,但其页面相关率均达到 13%以上,与传统爬虫相比,页面相关率提高了 9%以上。

本文实现了两个主题爬虫变种,并且根据收获率标准评价了三种主题爬虫的性能。尤其要强调的是 HMM 学习型爬虫,不仅学习目标页面的内容,而且还学习了导向目标页面的路径。从本质上说,网络蜘蛛的搜索问题是一个“多目标”规划问题。在合理的时间限度内,以较少的网络资源、存储资源和计算资源的消耗获得更多的主题相关页面是主题爬虫追求的最终目标。随着人们对“个性化”信息服务需要的日益增长,专业搜索引擎的发展将成为搜索引擎发展的主要趋势之一。

参考文献

[1] Zuo Xiaojun, Zhang Kaituo. An improved search algorithm of focused crawler in vertical search engine [C]. Asia-Pacific Youth Conference On Communication Technology2010 (APYCCT 2010), 2010: 509-513.

[2] Ju Xiaolin, Chen Jihong, Shao Haoran. Hierarchical Web page classification method based on vector space model[C].

Journal of Nantong University (Natural Science Edition), 2010.

[3] Yang Shengyuan. A focused crawler with ontology-supported website models for information agents [C]. Advances in Grid and Pervasive Computing, 2010:522-532.

[4] LI Jun, FURUSE K, YAMAGUCHI K. Focused crawling by exploiting anchor text using decision tree [C]. Proceedings of the 14th International World Wide Web Conference, 2005: 1190-1191.

[5] CHEN Y. A novel hybrid focused crawling algorithm to build domain-specific collections[D]. Ph. D. Thesis, Virginia Polytechnic Institute and State University, 2007.

[6] STEINBACH M, KARYPIS G, KUMAR V. A comparison of document clustering techniques [C]. Sixth ACM SIGKDD, World Text Mining Conference, Boston, MA, 2000.

[7] UDDIN M Z, LEE J J, KIM T S. Independent shape component-based human activity recognition via Hidden Markov Model[J]. Applied Intelligence, 2010,33(2):193-206.

(收稿日期:2010-11-01)

作者简介:

漆志辉,男,1984年生,硕士研究生,主要研究方向:数据挖掘与人工智能。

杨天奇,男,1961年生,博士,教授,主要研究方向:人工智能,神经网络,数据挖掘,搜索引擎技术。