

Web 网站死链检测方法

姚 卓, 蔡皖东, 姚 烨

(西北工业大学 计算机学院, 陕西 西安 710129)

摘 要: 网站作为大规模的信息集合体, 包含了大量的 Web 链接. 有些 Web 链接经过一段时间之后, 因种种原因而失效或者出现错误, 从而形成死链. 本文提出一种 Web 网站死链检测方法. 根据 Web 链接的调度过程, 自动获取网站链接信息; 根据 Web 链接的结构特点和网页检索操作, 对死链进行分析和检测; 针对链接的相互引用问题和用户体验与页面深度的关系, 对采集的数据进行预处理. 实验结果表明, 该方法能有效地提高死链的检测覆盖率和处理效率.

关键词: 网站; 死链检测; HTTP 协议; Web 链接

中图分类号: TP393

文献标识码: A

文章编号: 1000-7180(2012)12-0103-05

Website Dead Links Detection Method

YAO Zhuo, CAI Wan-dong, YAO Ye

(College of Computer, Northwestern Polytechnical University, Xi'an 710129, China)

Abstract: For large-scale information collection, Web sites contain considerable links. After a period of time, some web references that for a variety of reasons will not lead to a valid or correct web page, which is called dead link. The paper puts forward a dead link testing method. Achieve link messages automatically according to processes of URL dispatch. Analysis and detect dead links based on structural characteristic of web links and chain of actions needed to retrieve a web page; preprocess data collected before, due to a large number of relevant content links existed in a web page and relationship between user experience and page depth. Experiment results show that the method can improve coverage percentages and processing efficiency of dead links detection effectively.

Key words: website; dead link detection; HTTP protocol; web links

1 引言

死链是 Web 网站中处于路径不可达状态的 Web 链接, 也是导致网站性能下降的重要原因. 随着互联网的快速发展, 网络中的各种信息也在急速增长, 维护网络内容的完整性和维持网络的性能也变得十分复杂. 网络的一个重要特点是动态性, 即网页内容可以任意增加、删除和更新. 这个特点使网络成为具有吸引力的媒介, 同时也使它的完整性受到了威胁, 导致了死链的产生. 不仅如此, 多数链接是相互引用的, 再加上网络的频繁更新和删除, 使得死链问题更加严重.

文献[1-7]中, 提出检测和处理死链问题的一些方法. 但没有提供一种检测所有网络链接错误的方案, 只关注了客户端错误或者关联数据库的错误. 本文的贡献有以下三方面: ①根据 Web 链接的调度过程, 自动获取链接信息; ②具体解释了死链产生的原因并提出了对应的检测方案; ③对真实网络数据做了比较多的实验, 实验显示本方法具有较高的死链检测率.

2 相关概念

2.1 死链接定义

网络资源通常使用统一资源定位器(URL)来

收稿日期: 2012-04-13; 修回日期: 2012-05-14

基金项目: 西北工业大学基础研究基金(JC201149); 西北工业大学研究生创业种子基金(Z2012141)

表示. URL 是统一资源标识符 (URI) 的子集, URI 提供资源位置的抽象鉴定. 完整的、带有授权部分的普通统一资源标志符语法如式 (1) 所示^[8].

协议://用户名@密码:子域名.域名.顶级域名:
端口号/目录/文件名.文件后缀? 参数=值#标志
(1)

URI 中的服务器标识符可以有 HTTP://、FTP://、GOPHER://、TELNET://、NWES:// 等类型, 分别指定为采用超文本传输协议连接、采用文件传输协议连接、与 GOPHER 服务器连接、与 TELNET 会话连接、与 USENET 新闻组相连接, 在本文中的实验中, 由于 98% 的 URI 是通过 HTTP 协议传输的, 所以本文只考虑 http 协议组. 在 HTTP 方案下, 不需要用户名和密码. 主机使用完全限定的域名或者其 IP 地址. HTTP 端口的默认值是 80, 可以省略. ‘#’ 后面是可选部分, 它提供额外鉴定信息指向一个二级资源. 而 ‘?’ 之后的询问语句指向用户感兴趣的资源.

死链接也称无效链接, 即那些不可达到的链接. 以下情况会导致死链接的出现: 动态链接在数据库不再支持的条件下, 变成死链接; 某个文件或者网页移动了位置, 导致指向它的链接变成死链接; 网页内容更新并换成其他的链接, 原来的链接变成死链; 网站服务器配置错误.

2.2 死链接产生的原因及分类

要使一个给定的 URL 正常显示, 需要许多不同的技术和协议协同工作. 另外随着网络和互联网的不断变化, 原本简单的端对端请求应答协议变得十分复杂. 所以链接失效可能出现在检索 Web 网页的复杂操作链中的任何一环.

当网页首次请求时, 主机名必须先解析成有效的 IP 地址. 虽然 Web 出现之前, 非路由的 IP 地址和域名地址是一对一的关系, 但是现在这种对应关系却更加复杂了. 许多主机把不同网站和每个地址 (虚拟网络主机) 关联起来, 因此需要多个不同的 IP 地址. 由于 IP 地址逐渐变为有限资源, 所以通常情况下, 把许多域名关联到相同的主机和 IP 地址上来为不同的网站服务, 这些网站基于主机名来响应页面请求 (虚拟网络主机). 最后导致网站接收到大量数据, 这些网站为了把主机上的通信量分发出去, 会采用把不同的 IP 地址和主机名关联到相同的域名上的策略.

URL 中的文件路径不是必须和服务器上的相应的本地文件匹配对应的, Web 服务器提供了许多

机制管理名字空间, 包括: 为每个本地用户建立一个独立的空间, 保护域的定义和访问机制, 通过支持别名来把名字空间映射到本地目录中, 以及网页内容的动态生成——例如使用通用网关接口和动态服务器网页技术. 另外, HTTP 协议的一个特点是数据表示方式的可协商性, 该特性允许服务器根据请求消息里描述的特定技术或者文化特性来提供不同的页面 (例如带宽, 响应方式, 语言内容编码).

最后一个难题来自 Web 事务很少遵循用户主机到服务器的理想路径. 每个终端都被防火墙保护, 以至于网页内容经过防火墙时会被阻碍甚至是修改. 对于终端用户来说, 路由器使用网络地址解析机制来管理有限的 IP 地址, 该方法可能把真实 IP 地址屏蔽. 最后, 代理服务器——要么和终端用户合作, 要么公开拦截请求——可能在不和原始服务器协商的情况下, 存储缓冲文档并且为请求提供服务.

在以上复杂的情况下, 任何一步出错都会导致死链的产生. HTTP 协议已经定义了 24 种错误. 另外, 还有一些错误是协议没有定义的. 以下列举在实验中出现的常见错误.

(1) DNS 解析错误: 解析器递归查询完, 没有发现被请求信息的服务器地址, 也就是说没有把机器符号名映射到他们的 IP 地址上. 产生错误的原因可能是 DNS 服务器无法正常工作, 本机的 DNS 缓存列表没有更新, 或者是该域名没有注册.

(2) 重定向: 服务器已经把链接重定向, 但是没有指明后续地址. 这类链接的状态码以 3 开头, 表明客户应该检查其他的位置: 另一个不同的 URL, 或者在它自己的缓存中查找. 然而服务器并没有指明后续的链接地址或者在缓存中没有找到.

(3) 4XX 客户端错误

400 坏请求: 由于错误的请求不能被服务器理解, 在实际网站中, 这类错误主要由 URL 的错误键入引起的.

401 未授权的: 服务器需要对请求进行身份认证. 如果请求包含了授权证书, 那么 401 响应意味着对这些证书的授权失败. 通常, 这样的错误是由于引用了需要注册的域名服务器里的 URL. 但是把返回码为 401 的链接看做错误链接具有争议性, 因为越来越多的网站服务需要注册授权, 但不意味着网站的链接是无效的.

403 禁用: 服务器理解此请求, 但拒绝满足该请求. 认证是没有作用的, 并且请求不能被重试. 例如, 学校的校园网对学生是透明的, 这种验证是基于 IP

地址的验证,但对其他用户是禁止的,并提示 403 错误。

404 没有找到:服务器没有找到任何可以匹配的资源,并且没有迹象表明条件是暂时的还是永久的。当服务器不希望精确指出请求为何被拒绝或者没有任何其他响应可用时,使用该状态码。该错误产生的原因一般是由于网站维护人员改变了资源的路径或者删除了整个引用资源。

410 不存在:请求资源在源服务器上不再可得并且也没有转发地址可用,此条件被认定是永久的。

(4)5XX 服务器错误

500 服务器内部错误:该错误产生的原因是服务器被错误配置或者支持动态页面的程序或数据库运行失败。

503 服务不能获得:由于服务器暂时地过载或维护,服务器不能处理请求。该状态码并不总表示服务器过载,有些服务器可能希望拒绝此连接。

504 网关超时:即网关或者代理服务器不能及时接收从 URI 指定的上游服务器(例如:HTTP,FTP,LDAP 服务器)或者其他的辅助性服务器(DNS 服务器)发来的响应。当 HTTP 请求被代理缓存服务器拦截时,也会出现该错误。

(5)没有返回有效内容:应答正常返回,但是网页内容为空。可能是由于网站维护,服务器原因或者用户的防火墙篡改。

3 死链接检测方法

本方法检测步骤如下:

- (1)爬取网页内容,并保存完整页面信息;
- (2)把页面中的链接抽取出来;
- (3)对这些链接进行预处理;
- (4)检验链接的可达性。

3.1 链接自动获取

为了完成死链的自动检测,首先需要实现链接的自动发现和提取功能。本文以 Web 页面之间的相互链接为路径,依照一定的遍历策略试图获得网站中的每个链接,从中抓取必要的信息,并将其转存到本地存储设备上,以便进行信息的预处理。有两种经典遍历策略,深度优先和广度优先,其目的是尽量获取较多的网页。本文根据 URL 的不同调度状态和即将进行的操作对传统的广度优先遍历策略进行修改。

算法首先初始化一个种子列表——seedURLs 和五个 URL 列表:用来存放 DNS 缓存的 Named-

Site;用来存放已取得 DNS 解析的 URLs 的列表 IPSite;用来存放没有进行 DNS 解析的站点 dns-Site;存放即将抓取的 URL 的 okSite 以及用来存放死链的 errorSite。然后进行如下操作,如图 1 所示。

第一步:从 seedURL 中选择一个 URL,把它放在 NamedSite 列表中,如果这个 URL 需要 DNS 解析,把它放在 dnsSite 列表中,进入步骤 2,否则进入步骤 3。

第二步:解析 URL,计算死链值,如果为 0,进入步骤 3,否则放到 errorSite 列表中。

第三步:从 okSite 中选择一个 URL,发送请求到相应的服务器,根据返回的头信息和内容计算死链值,如果值为 1,将该 URL 加入到 errorSite 列表中。

每个站点都用 namedSite 来保存自己站点的 URL,然而这个个数是有限的,当超过这个限制的时候,就不能再把此站点下的 URL 放入,而是放入等待队列。算法会控制一段时间在 URL 库中取数据,一段时间在等待库中取数据。

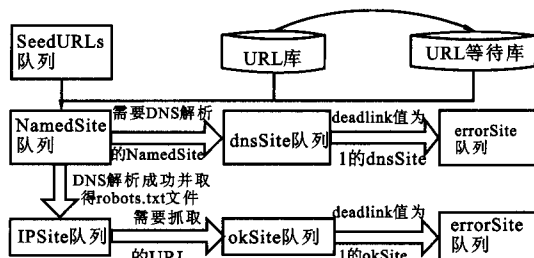


图 1 URL 队列管理

3.2 数据预处理

预处理主要包括网页去重和链接筛选,如图 2 所示。上一步得到的 URL 列表含有大量的重复项和不符合深度约束的链接,为了提高计算效率、减少存储空间,需要首先对其进行去重处理,然后删除不符合约束的链接。



图 2 预处理过程

Web 可以看成是一个没有规则和边框的有向图,图的内部存在一个广大的圆圈。由于网站中含有大量的索引页、导航页面,以及大量的相关链接,网页中的 URL 数目远远超过互联网中存在的实际 URL 数目。为了避免下载重复页面,首先需要考虑 URL 过滤。Bloom filter^[9]是一个改进的算法,具有高效性和占用内存资源少的优势,因此系统采用 bloom

filter 算法来过滤重复链接. 对于每一个 URL, 算法输出唯一的 MD5 值, 具体如下:

```
Hash method()
{
    h0 = port;
    While(host is not null){
        h(xi+1)=PF*h(xi)+f(xi);
    }
    While(file is not null){
        h(xi+1)=PF*h(xi)+f(xi);
    }
    Code=h(xn)% hashSize;
}
```

在上述算法中, PF 是一个质因子, $h(x_i)$ 是每次计算产生的随机函数, $f(x_i)$ 是字符集, 其中包含等待映射的字符. Code 就是该 URL 对应的 MD5 值.

为了达到数据采集效率和采集完整性的平衡, 本文采用以网页深度作为条件过滤 URL 的方法. 网页深度是指从一个网站的主页, 到达网站的某一网页所需要经过的最短跳转次数. 据大量实验统计表明, 如果只有一台服务器做数据采集工作, 那么当网页深度为 0 时, 链接重要程度是最高值 10, 链接个数为最小值 1; 当深度增加到 3 时, 平均链接个数增加到 2000, 但是重要程度下降到 2; 当深度增加到 4 以上时, 链接增长率开始下降, 重要程度也迅速递减. 该研究表明把深度设为 5 时, 服务器基本上可以获得网站中的所有链接而不需要把过多时间浪费在处理重复页面上. 虽然删除网页深度过大的 URL, 可能造成死链的漏报, 但并不影响用户体验. 因为据 Google2010 年 1 月对 Google Analytics 的基本度量——访问时长和访问深度的大量实验中, 可以看出大约 80% 的用户不再访问深度 > 5 的页面, 如图 3 所示.

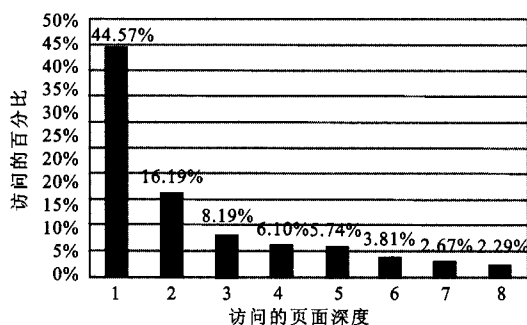


图3 深度与用户体验的关系

通过上述处理, 在保证 URL 链接完整性的前提下, 使原始 URL 库得到了很大的精简, 同时系统的整体性能得到了很大的提高.

3.3 死链分析和判断

经过预处理操作得到最精简的 URL 库, 为了找出死链, 需要对其中每一个 URL 做进一步的分析 and 判断. 算法思想: 模拟浏览器作为客户端代理向 DNS 服务器以及网站服务器发送询问请求, 根据第 2 部分对死链的产生的原因, 分类以及 HTTP 协议的定义, 分析返回的信息, 若消息满足死链特征, 则检测到死链, 做相应处理. 如图 4 所示, 具体算法如下:

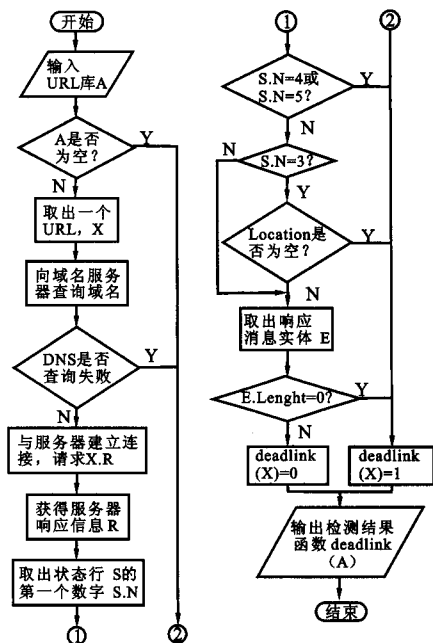


图4 检测算法流程图

算法名称: 死链检测算法

算法输入: 预处理后 URL 库 A

算法输出: 死链检测结果函数 $deadlink(A)$

算法步骤:

(1) 如果 URL 库 A 非空, 取出一个 URL X; 如果 A 为空, 算法结束, 死链判断完毕.

(2) 向 DNS 服务器发送链接 X 的域名查询请求;

(3) 域名服务器通过递归查询返回链接 X 的 IP 地址 X. IP, 如果递归查询失败, 则 $deadlink(X)=1$, 返回步骤 1;

(4) 与 X. IP 建立一个 TCP 连接后, 向服务器发送消息请求, 请求资源 X. R, 请求消息中包括 GET 方法, 资源指示符, 请求头域以及实体头域;

(5) 服务器接收和解析一个请求消息后, 发出一个 HTTP 响应消息 R, 响应消息中包含状态行 S、响应头域以及实体 E;

(6) 取 S 的第一个数字 S. N, 如果 S. N=4 或者

5,则 $deadlink(X)=1$,返回步骤 1;如果 $S.N=3$,查看响应头域的 Location 字段,如果内容为空或者是不合语法的 URL,则 $deadlink(X)=1$,返回步骤 1;

(7)取实体 E 进行分析,如果实体的长度 $E.Length=0$,则 $deadlink(X)=1$,返回步骤 1;

(8) $deadlink(X)=0$,返回步骤 1.

4 实验结果及分析

本实验运行在 Linux 操作系统、2 GB 内存、Intel Core2 2.53 GHz 处理器的主机上,并连入互联网,对方法的采集性能和检测精度进行测试.

实验用两种目前流行的技术:一个简单的网页版本(站长帮手网死链检测网页)和一个 windows 平台上的软件(Xenu),同本文提出的跨平台检测方法作对比.其中参数之一的检测精度,需要通过人工手动检测正确性.表 1 显示了不同的技术在检测 $www.nwpu.edu.cn$ 时的不同性能.为了保证比较的公平性,实验设定 6000 为阈值,所有版本都在处理了 6000 个页面后停止.对表 1 的分析可知,网页检测只处理了 31 个链接,原因就是网页检测方法只抽取初始页面的超链接做后续处理,不对生成的 URL 库做进一步分析从而得到后续 URL.表 1 中后两个参数表明,即使后两种技术有相同的正确率,但是本文提出的方法能检测出更多的死链.

表 1 性能比较

参数 技术	处理的 URL 数	检测出 的死链	正确的 死链数	查准率
网页版本	31	1	0	0
Xenu	6000	14	14	1
本方法	6000	57	57	1

为了充分证明本方法的性能优越性,需要设计大量的实验进行对比.首先随机选取 12 个具有代表性的网站,确保每个网站的类型各不相同,其次为了保证实验的准确性和公平性,需要分别对每个网站设置不同的停止阈值,最后使用两个系统对这 12 个网站进行测试.实验结果如图 5 所示,横坐标表示处理的链接个数,纵坐标分别表示检测出的死链个数和正确率.其中,柱状线表示检测出的死链个数,折线表示正确率.图 5 的统计结果表明,本方法在采集性能和检测精度上,都有较大的改进.

5 结束语

本文针对 Web 链接的结构以及检索 Web 网页的复杂操作,解释了死链产生的多种具体原因,并针

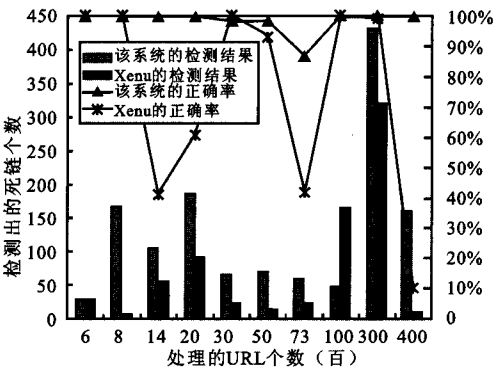


图 5 性能比较

对不同原因提出了对应的检测方案.根据 Web 链接的调度过程,对链接获取技术进行了有针对性的修改,从而提高了链接采集性能和死链检测的精度和覆盖率.

为了进一步提高检测率,今后将采用基于语义的方法来改进 Web 网页获取技术,实现对网络的深度挖掘.

参考文献:

[1] Haslhofer B, Popitsch N. DSNotify- detecting and ? xing broken links in linked data sets[C]// DEXA 09. 20th International Colocated with DEXA. Austria, Linz; IEEE, 2009:89-93.

[2] Popitsch N, Haslhofer B. Dsnotify: handling broken links in the web of data[C]// Proc of WWW. New york, NY, USA: ACM, 2010:761-770.

[3] Niko Popitsch, Bernhard Haslhofer. DSNotify-A solu- tion for event detection and link maintenance in dy- namic datasets[J]. J Web Sem, 2011,9(3):266-283.

[4] Harrison T J, Nelson M L. Just-in-time recovery of missing web pages[C]// Proc of HYPERTEXT. New York, NY, USA: ACM, 2006:145-156.

[5] Vesse R, Hall W, Carr L. Preserving linked data on the semantic web by the application of link integrity techniques from hypermedia[C]// LDOW2010, Colo- cated with WWW ' 10. Raleigh, North Carolina, USA; IEEE, 2010.

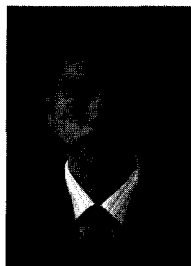
[6] Morishima A, Nakamizo A, Iida T, et al. Bringing your dead links back to life: a comprehensive approach and lessons learned[C]// Proc of HT New York, NY, USA: ACM, 2009:15-24.

[7] Klein M, Nelson M L. Evaluating methods to rediscover missing web pages from the web infrastructure [C]// Proc of JCDL. New York, NY, USA: ACM, 2010:59-58.

(下转第 111 页)

- 2010,77(5):898-916.
- [7] Chen Q, Yang O W W. Robust controller design for AQM router[J]. IEEE Transactions on Automatic Control, 2007,52(5):938-943.
- [8] Zhang Y, Kang S R, Loguinov D. Delay-independent stability and performance of distributed congestion control[J]. IEEE/ACM Transactions on Networking, 2007,15(5):838-851.
- [9] Zheng F, Nelson J. An H_∞ approach to the controller design of AQM routers supporting TCP flows[J]. Automatica, 2009,45(3):757-763.
- [10] 张长旺,殷建平,蔡志平,等. 抗 DDoS 攻击的主动队列管理算法[J]. 软件学报,2011,22(9):2182-2192.
- [11] Manfredi S, Bernardo M D, Garofalo F. Design, validation and experimental testing of a robust AQM control[J]. Control Engineering Practice, 2009 (17): 394-407.
- [12] Kim B. Tail asymptotics for the queue size distribution in a discrete-time Geo/G/1 retrial queue [J]. Queueing System, 2009(61):243-254.
- [13] 田乃硕,徐秀丽,马占友. 离散时间排队论[M]. 北京: 科学出版社,2008.

作者简介:



段谟意 男,(1964—),副教授. 研究方向为计算网络.

(上接第 107 页)

- [8] Wenqing yuan. Research on Prototype Framework of a Multi-threading Web Crawler for E-commerce[C]// Management and Service Science(MASS). China, wuhan;IEEE,2009:20-22.
- [9] Xiao-Guang Liu, Jun Lee. K-divided bloom filter algorithm and its analysis[C]//Proc of FGCN. Washington, DC, USA: IEEE Computer Society, 2007: 220-224.

作者简介:

姚卓女,(1990—),硕士研究生. 研究方向为网络信息安全.

蔡晓东 男,(1955—),教授,博士生导师. 研究方向为网络信息安全.

姚焯 男,(1971—),副教授. 研究方向为网络信息安全.