Journal of Computer Applications
计算机应用,2013,33(2):430-433
ISSN 1001-9081
CODEN JYIIDU
2013-02-01
http://www.joca.cn

文章编号:1001-9081(2013)02-0430-04
doi:10.3724/SP.J.1087.2013.00430

基于局部变化性的网页篡改识别模型及方法

魏文晗[1*],邓一贵[2]
(1.重庆大学 计算机学院,重庆 400044; 2.重庆大学 信息与网络管理中心,重庆 400044)
(*通信作者电子邮箱 wenhanwei@cqu.edu.cn)

摘 要:针对传统的网页远程监控方式局限于静态网页的问题,提出一种适用于动态网页的基于规则的分类模型。该模型考虑到网页的局部变化性,首先根据历史页面的动态更新,划分网页的动态区域和静态区域;其次,对动态区域,根据历史特征计算相关阈值,对静态区域建立分块的MD5历史库;最后,根据定义的IF-THEN规则决定是否发送警报。实验表明,该模型能在更短时间内完成全站检测,对正常页面的误报率较低,对异常页面的检测率较高。

关键词:网页篡改;网站监测;篡改检测;IF-THEN规则;领域知识
中图分类号:TP393.08 **文献标志码**:A

Detection model and method of website defacements based on attributes partial changes

WEI Wenhan[1*], DENG Yigui[2]
(1. *College of Computer Science, Chongqing University, Chongqing* 400044, *China*;
2. *Information and Campus Network Management Center, Chongqing University, Chongqing* 400044, *China*)

Abstract: The traditional methods of website remote monitoring are limited to static webpages. A rule-based classifier for dynamic webpage was proposed. The method took the website partial changes into consideration, and divided the websites into the dynamic regions and the static regions according to the dynamic updates of the historical pages, and then calculated thresholds based on the historical features for dynamic regions and built history database of MD5 based on blocks for the static regions. Finally, it decided whether to send alarms according to the defined IF-THEN rules. The test results show that the model can scan the whole website in shorter time, get lower false detection rate for normal pages and higher detection rate for distorted pages.

Key words: Web defacement; website monitoring; defacement detection; IF-THEN rule; domain knowledge

0 引言

网页篡改是目前较为普遍的一类网站攻击形式,它是指黑客利用特定手段入侵网站,将网站内容部分和完全替代。根据CNCERT/CC最近的报告[1],中国大陆地区2011年被篡改网站与2010年相比增加了5.1%,其中政府部门、公益组织、教育机构网站容易成为黑客篡改的目标。因为网站整体安全性差,缺乏必要的安全性维护,很多网站被篡改后长期无人过问。文献[2]对62000个被篡改页面连续两个月的观察发现:43%的网站在1周后才对篡改事件做出响应,而超过37%的网站在两周后,依然未发现页面被篡改。页面被篡改不仅导致正常的业务无法运营,更损害了政府或公共机构的形象,有些站点甚至被利用间接成为非法牟利的工具。

1 相关工作

网页篡改检测的目标是在页面文件被篡改后,能够及早发现,并通知管理员应急处理。根据安装位置不同,目前检测方法主要分为两类:基于网站服务端的本地检测和基于客户端的远程检测。基于服务端检测多采用核心内嵌或文件过滤驱动技术[3-5]对每个流出网页或磁盘I/O进行完整性检查或者过滤。此类技术精度高,而且能做到实时防护,不足之处是需要在每个服务端上安装专门的软件,不能做到大规模的篡改检测。除此之外,它让管理员操作变得复杂,并且占用服务器的系统资源,降低网站性能。

基于客户端的远程检测只需知道目标网站的域名即可,适合大规模的检测。它主要利用网页爬行技术对抓取的页面进行分析。文献[6-9]是早期经常被使用的方法,它通过计算页面的MD5值,并将页面当前版本的MD5值与上一版本进行比较,如果发现不同则说明网页被篡改。该方案比较适用于静态页面,而目前动态页面技术应用越来越广泛,此方法对绝大多数网站都已失效。针对动态页面,网页篡改检测的难点是如何能正确分辨页面的变化是正常更新的内容还是篡改的内容。文献[10-11]借鉴核心内嵌的思想,首先基于安全(如加密等)传输技术从Web服务器处获得未发布的页面并建立Web页面基线,然后通过分析页面的文档对象模型生成Web页面的页面模式,并将页面模式与基线数据库的页面模式进行比较;文献[10-11]虽然基于远程分析,不过建立基线数据库时依然需要Web服务器的配合,不适合大规模检测。文献[12]引入了*n*-gram模型,将整个HTML文件看作字符串,然后计算相似度;不过文献[12]对每个页面都要进行一段时间的阈值训练,即一个页面对应一个训练阈值,如果网站新增的页面本身可疑,该方法无法做出识别。文献[13]提出了一套远程监视服务的框架,其主要思想是网页篡改必然会改变网页文件的基本属性,比如改变页面编码格式、页面字节大小、行数量、文本块大小等;改变HTML标签元素的特征分布,出现上一页面没有的HTML标签元素、不同标签类型数量异常比如表单类型FORM、TEXTAREA、LABEL等,部分HTML标签元素在整体上或者DOM树的某一行上的频率出

收稿日期:2012-08-03;**修回日期**:2012-09-16。 **基金项目**:重庆市自然科学基金资助项目(CSTC2011JJA40023)。
作者简介:魏文晗(1986-),男,湖北天门人,硕士研究生,主要研究方向:计算机网络、信息安全; 邓一贵(1971-),男,四川简阳人,高级工程师,博士,主要研究方向:计算机网络、信息安全、移动代理。

现大的波动等;或者引起 HTML 风格出现异常,包括字母大小写切换比率、数字字母切换比率等。基于此思想,文献[13]首先把待特征提取的目标划分为四大类,共计 1 466 个特征,然后将在正负样本集中提取到的特征向量在已有的分类算法中进行分类训练。该方法基于整体策略,弥补了文献[12]的不足,并且精度有所提高,但是它没有从内容篡改上考虑,而且由于它选择的特征很多,每次检测的时间成本非常高。

针对上述问题,本文提出一种实时大规模的远程篡改检测方法。该方法基于网页局部变化的特性,对网页布局和内容进行检测,不仅检测精度高、时间开销少,而且能对可疑位置进行定位。

2　问题定义

2.1　形式化定义

网页文件主要由 HTML 标签和文本内容组成。网页篡改引起的异常变化也主要体现在 HTML 标签和文本内容的不同上面。为了信息发布的方便,一般动态网站都是基于后台来定制和更改前台页面的内容和样式。定制的内容和样式由动态语言来实现,通常称为网页模板。因此对网页篡改内容的识别,主要是通过分析 HTML DOM 树挖掘出页面特征,并将当前页面特征与历史特征进行比较,如果差异触犯给定的规则,则报警。

为了后文叙述的方便,本文先对 HTML DOM 和 DOM 树之间的相似度进行形式化定义。

定义 1　HTML 文档对应的 DOM 树定义为 T,$T=\{L_1,L_2,\cdots,L_m\}$,其中 $L_i(i=1,2,\cdots,m)$ 代表第 i 个的节点,它是一个五元组,$L_i=(F,RC,ATT,CT,ST)$,这里 F 代表从根节点到当前节点的路径;RC 代表当前节点在 DOM 树的第几层第几个,比如 23 代表该节点在 DOM 树的第 2 层第 3 个;ATT 代表当前节点的样式;CT 代表当前节点与下一节点之间的文本内容,如 <a>tatb 的子树,节点 <a> 中 CT 的值应为 ta;ST 代表节点状态,分为动态、静态和未知三类。

定义 2　DOM 树 T_1 与 T_2 的相似度:

$$Similarity(T_1,T_2)=\frac{2*|T_1\cap T_2|}{|T_1|+|T_2|} \tag{1}$$

其中:$|T_1|$、$|T_2|$ 分别代表两棵树节点的数目,$|T_1\cap T_2|$ 代表两棵树相同节点的数目。

两棵树节点相同,分两种情况:

1)两棵树结构相同,即所有节点 L_i 中的 F 和 RC 属性值相同,称为结构相似;

2)两棵树结构相同并且内容也相同,即所有节点 L_i 中的 F、RC、CT、ST 属性值相同,称为内容相似。

2.2　网页局部变化性实验及分析

为了方便信息浏览和观赏体验,网页设计者通常会把网页大致分成三个部分:导航栏、主题区域和版权信息。因此网页通常是由一个个信息块组成。同时页面的更新也是以一个个网页块的形式进行的,更新时页面上大部分内容并没有变化[14-15]。如果在特征提取的过程中,依然把整个页面当作处理单位,则会影响网页处理的效率。

网页的局部变化性包含两层含义:1)页面的更新方式是局部更新;2)一个网站在一定时期内只有少量的页面经常会更新,大部分页面一旦生成后便不再变化,内容经常更新的链接之和远小于链接总数,即一个页面内容生成后几乎不会被修改。需要注意的是,该特性不考虑网页被移除的操作,即网页不可访问,而只考虑网页的更新和修改操作;这里的时间约束虽然是在一定时期内,但是可以将网页整体样式的更新认为是一个很长的周期。通常整体样式的更新会造成所有页面 DOM 结构和页面样式的改变,但这种操作不经常发生。

为了验证网页局部变化性,本文选取 100 个网站,每隔一个小时对页面抓取一次,为期一个月。按照式(1)对所搜集到的页面进行内容相似度计算发现:网站内动态页面与静态页面(这里特指每次计算相似度均大于 0.90)的链接之比平均值为 0.014。图 1 显示比率从大到小的分布图。从图 1 可以看出:网站中只有很少的页面才是动态更新,大部分页面一旦生成,内容便不再变化。

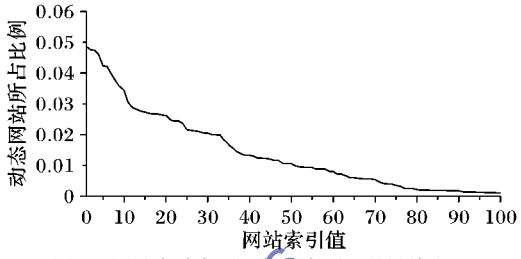

图 1　网站内动态页面与静态页面的链接之比

另外,本文将网站首页在各时间点获取的版本与最初的版本按照式(1)计算内容相似度发现:大部分网站在监测一段时间后进入稳定期,表明动态更新区域已经全部变化。从图 2 中可以发现:页面中只有一部分 HTML 标签的内容在变化即局部更新方式,并且更新的过程是一个随时间渐变的过程。

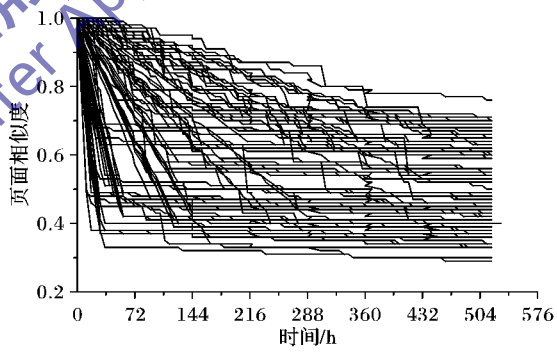

图 2　100 个页面相似度的曲线分布图

2.3　网页篡改的特征与分析

根据对网页篡改案例的总结,可以发现篡改内容具有以下几个特征:

1)被植入恶意的 HTML 脚本,即页面被加入恶意的 <script>…</script> 标签。

2)发布虚假信息,主要分为两种:篡改已有信息和添加新的动态内容。根据 2.2 节可知网页内容一旦生成后便很少修改,因此本文认为所有的修改内容都为篡改内容。对于新生成的内容,由于基于语义分析也很难判断是否为合法内容,因此本文只对异常特征进行识别,比如添加内容长短、添加时间异常、字体异常、敏感内容检测等。

3)网页被替换或被修改,这里指网页布局发生变化,主要体现在 DOM 结构上的变化。

3　网页篡改识别模型

网页篡改识别模型的整体流程:首先在训练期区分静态页面和动态页面,并提取动态页面的静态区域和动态区域,同时对静态页面和动态页面的静态区域建立 MD5 值的历史库;在监测期,对静态页面和动态页面的静态部分与历史库中的 MD5 值进行比较。对于动态页面首先比较结构相似性,再根据表 1 的先验规则对页面布局和内容进行检测;对于新增页

面,先按照 URL 相似性进行粗粒度聚类,然后按照结构相似性进行细粒度聚类,如果为静态类且不符合表 1 的先验规则,则建立分块 MD5 值,否则将该链接提交给监测端。

表 1 IF-THEN 规则表

页面类型	规则 IF 条件
所有页面	规则 1:不包含任何标签[13]; 规则 2:不包含任何可见文本[13]; 规则 3:仅包含一张图片或根本没有图片[13];
静态页面	规则 4:提取 MD5 历史库中该页面各块的 DOM 位置,如果没找到或者当前计算的 MD5 值与历史不同;
动态页面	规则 5:提取 MD5 历史库中该页面静态区域各块的 DOM 位置,如果没找到或者当前计算的 MD5 值与历史不同; 规则 6:与上一版本的结构相似度少于阈值 th_1; 规则 7:内容发生变化的标签数量大于阈值 th_2; 规则 8:训练期结束后,动态页面发现未知 HTML 标签; 规则 9:新增内容的长度小于阈值 th_3; 规则 10:新增内容的字体大于所属分类最大字体的 th_4 倍;
新增页面	规则 11:没有找到与该页面 URL 相似的 URL 分类; 规则 12:如果规则 6 为 FALSE,没有找到与该页面结构相似的分类

3.1 提取页面静态区域和动态区域

每次从远端获取的页面如果与上一次获取的页面不同,该页面的版本号就会累加。通过对连续版本的比较分析就可以提取出页面的静态区域和动态区域。该过程的形式化定义如下。

定义 3 若 DFF 为同一页面当前版本与上一版本不同的所有节点,则

$$DFF_{i+1} := T_{i+1} - T_i \cap T_{i+1} \tag{2}$$

定义 3 可由定义 2 推导出,这里证略。

定义 4 若 DA_i 为时间点 i 得到的动态区域,则

$$DA_{i+1} := DA_i \cup DFF_{i+1} \tag{3}$$

从图 2 中可以发现不同的页面进入稳定期的时间不同。如果直接使用式(3)提取动态区域,有些页面在 12 h 内可以完成动态发现,而有些页面却需要几周以上。经过对多所大学的校园网站进行分析,发现动态变化的标签通常是排成一行或者一列,该标签的路径上至少有以下三种标签的一种:<TABLE>、<DIV>、<UL>。基于此,本文提出如下改进算法:

输入:当前页面 DOM 树 T_{i+1},上一页面 DOM 树 T_i,上一页面得到的动态节点集合 DA_i;

输出:动态节点集合 DA_{i+1}。

算法步骤:

```
Begin
1) 初始化 DA_{i+1}为空;
2) 初始化节点队列 List 为空;
3) 依据式(2)、(3)计算 DA_{i+1};
4) 将 DA_{i+1}中每个节点赋给 List;
5) While(List 不为空)
     取出 List 最后节点 L;
     If(L.ST 为静态) Then
       If(L 的路径 F 包含"ul") Then
         将离 L 最近的"ul"节点加入队列 List,并将其所有子节点状态 ST 修改为动态后添加进 DA_{i+1}中;
       End If
       If(L 的路径 F 包含"table") Then
         找到离 L 最近的"table"节点将其所有子节点状态 ST 修改为动态后添加进 DA_{i+1}中;
       End If
       If(L 的路径 F 包含"div") Then
         找到离 L 最近的包含样式的"div"节点将其子节点状态 ST 修改为动态后添加进 DA_{i+1}中;
       End If
     Else
       Continue;                                    //取下一个节点
     End If
   End While
End
```

所有的网页初始都被认为是静态网页。在训练期如果发现网页出现变化,通过前面算法得到动态区域后,剩下区域都为静态区域。如果连续 th_0 天页面都没发生变化,训练期结束。th_0 即为训练时间,该时间与网站的更新频率有关系。

3.2 静态网页和静态区域识别

为了达到内容识别和篡改位置的定位,本文改进传统的 MD5 签名方式,采用分块的 MD5 签名策略。如下所示:

```
<!--Signature: dfhdfj213jh-->
<DIV>
  文字...
</DIV>
<!--Signature: werrew324uio-->
<DIV>
  文字...
</DIV>
```

首先设定最大信息量,对于静态区域,如果该区域所有节点的文本内容小于最大信息量,则直接计算该区域 MD5;如果大于最大信息量,则将该区域分块,直到每块的信息量小于最大信息量。最后将每次计算的 MD5 值按照(起始节点,结束节点,MD5)保存下来。比如(23,27, werrew324uio)表示 DOM 树第 2 层第 3 个节点到第 2 层第 7 个节点的之间所有文本的 MD5 值为 werrew324uio;(33,33, dfhdfj213jh)表示 DOM 树第 3 层第 3 个节点所有子节点的文本内容为 dfhdfj213jh。

对于静态网页,除了分块计算 MD5 值,还会计算全局的 MD5 值。如果全局的 MD5 值不同,则顺序比较块 MD5 值,一旦发现异常,将块位置及异常信息发送到监测端。

3.3 根据 URL 聚类

一般情况下,同一 URL 目录中的网页大多来自于同一模板,其结构的相似性也很高。根据 URL 相似度进行聚类,可提高篡改检测的时间,节省对新增网页的训练时间。

设 f,g 表示两个网页的 URL 地址,则 URL 相似度的计算公式如下:

$$Url\_sim(f,g) = \frac{\max(1,|f \cap g|)}{\max(1,\max(|f|,|g|) - 1)} \tag{4}$$

其中:| | 表示字符串长度,∩ 表示从起始位置开始的公共字符串。

3.4 根据领域知识的网页篡改识别

由文献[10]的实验结果可知:领域知识聚合器比单纯的使用机器学习算法更能提高篡改检测的正确率和降低误报率。为了增强篡改检测的精度,本文基于对实际案例的分析和实验总结,提取出如表 1 所示的 IF-THEN 规则表。其中每个规则都是一个布尔类型,当规则 IF 条件为 TRUE 时,则向

监测端发送警报。

在监督学习的环境下,阈值的设置直接决定系统最终检测的准确率和误报率。由于篡改行为无法预测,因此阈值的设定更多是依赖于经验值。比如对于规则 9,可以将 th_3 设定为历史内容的最小长度;也可以根据历史内容的长度分布,选择一个置信下限。

4　实验及分析

本文从 2.2 节描述的 100 个网站中随机选取 20 个网站,每隔 4 h 对整个网站爬行并下载页面,为期两个月。将每个网站每次下载的页面按采集时间进行排序,平均每个页面建立了 240 个序列。根据页面更新频率,本文将 20 个网站分为 3 组(hour、day、week),更新频率分别在半天左右、一天左右、一周左右。根据搜索的历史记录,上述三组分别有 5、8、7 个网站。本文从公开的网页篡改档案库 zone-h. com 上搜集实际篡改页面 20 个。在每个网站第 12、24、36、48、60 天的采集序列中,用搜集的篡改页面随机选取页面进行取代,同时从每个网站剩余页面中随机选取 80 个页面进行手工篡改,分别对文本内容、图片、HTML 标签进行增加或修改,添加或删除脚本等,共建立 10 000 个篡改页面。

将所有网站序列托管到远程的 Web 服务器模拟运行,以轮转的方式依次访问每个页面的 240 个序列,模拟篡改监测。

在实际结果的衡量中,本文采用二值分类器常见的两个指标假正率(False Positive Rate,FPR)、假负率(False Negative Rate,FNR),在这里定义如下:

FPR = 被认为正常的篡改页面数 / 篡改页面总数

FNR = 被认为异常的正常页面数 / 正常页面总数

实验中训练时间设置为 40 次访问序列时间,其他阈值采用均值假设检验法进行自动调整。为了更好地验证本文提出的方法(简称为 PCR)的有效性,将该方法与文献[9]的 2-gram、文献[10]的 DomainKnowledge(简称 DK)进行了比较。

由于 2-gram 算法通过字符串比较发现异常,从图 3 可以看出其检测准确率最高,比 PCR 平均高出 5 百分点,不过 PCR 比 DK 高出了近 8 百分点。随着时间的增长,由于网页的变化造成参数调整,PCR 和 DK 的 FPR 值稍微有所上升,但影响不明显。同样,从图中也可以看出三者检测准确率受网页变化频率的影响并不明显。因此,在网站样式不出现大的调整的情况下,网页变化频率和时间长度对算法影响不大。

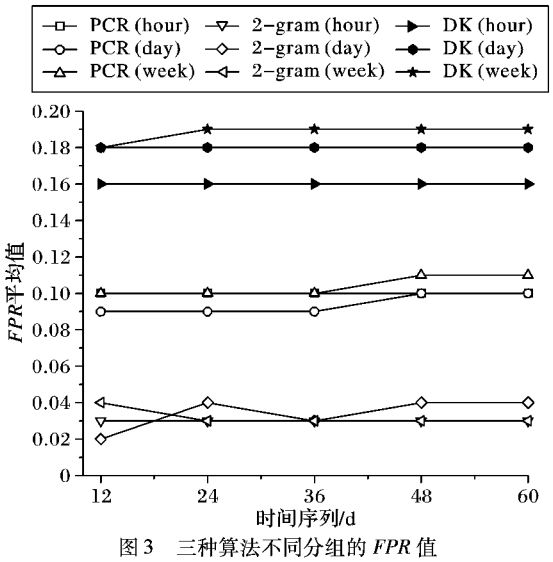

图 3　三种算法不同分组的 FPR 值

在实际运行中,系统返回的警告和运行时间也是一个衡量标准。本文对 3 个算法在整个运行过程中的 FNR 和完成时间进行比较。由表 2 可以看出 2-gram 的运行时间高,且警告数很多,PCR 运行时间最短。由于篡改事件对于一个网站而言并不是频发事件,因此过多且无细节说明的警告是无意义的。因此,PCR 算法最适合在现实中运用。

表 2　三个算法的 FNR 平均值和运行时间

算法	FNR 平均值/%	运行时间/h
PCR	12.03	25.0
DK	9.24	34.3
2-gram	43.77	138.6

5　结语

本文充分地考虑了网页的局部变化性,提出了一种适合于大规模远程扫描的网页篡改检测模型,并通过静态和动态的分开处理,提高了检测性能,缩小了定位异常的区域。同时,该模型也存在不足之处:1)容易受网页改版的影响,不能在短时间识别这种行为,而返回大量警告;2)当前对 JS 脚本检测较差,仅当作静态内容处理,还缺乏深入分析。

参考文献:

[1] 国家互联网应急中心. 2011 年中国互联网网络安全报告[EB/OL]. [2012 - 05 - 23]. http://www. cert. org. cn.

[2] BARTOLI A, DAVANZO G, MEDVET E. The reaction time to Web site defacements [J]. IEEE Internet Computing, 2009, 13(4): 52 - 58.

[3] 张建华, 李涛, 张楠. Web 页面防篡改及防重放机制[J]. 计算机应用, 2006, 26(2): 327 - 328.

[4] 孔辉. 一种网页防篡改系统的设计与实现[D]. 北京: 北京邮电大学, 2011.

[5] 阮宏伟, 李华, 王小雨, 等. 基于快照轮询和文本检测的批量网页防篡改系统[J]. 广西大学学报: 自然科学版, 2011, 36(A01): 142 - 147.

[6] 刘宝旭, 许榕生, 齐法制. 网站实时监控与自动恢复系统[J]. 信息网络安全, 2005(9): 69 - 71.

[7] 高延玲, 张玉清, 白宝明, 等. 网页保护系统综述[J]. 计算机工程, 2004, 30(10): 113 - 115.

[8] 白建坤, 张玉清. Linux 下网页监控与恢复系统的设计与实现[J]. 计算机工程与设计, 2006, 27(24): 4619 - 4621.

[9] TUSHAR K, VINEET R, VIVEK R. Implementing a Web browser with Web defacement detection techniques [J]. World of Computer Science and Information Technology Journal, 2011, 1(7): 307 - 310.

[10] 北京启明星辰信息技术股份有限公司. 一种 Web 网页篡改识别方法及系统: 中国, 201010034272. 5[P]. 2011 - 07 - 20.

[11] 赵帮, 何倩, 王勇, 等. 基于 LZMA 和多版本的网页防篡改备份恢复机制[J]. 计算机应用, 2012, 32(7): 1998 - 2002.

[12] KIM W, LEE J, PARK E, *et al*. Advanced mechanism for reducing false alarm rate in Web page defacement detection [C]// WISA'06: The 7th International Workshop on Information Security Applications, LNCS 4298. Berlin: Springer-Verlag, 2006.

[13] DAVANZO G, MEDVET E, BARTOLI A. Anomaly detection techniques for a Web defacement monitoring service[J]. Expert Systems with Applications: An International Journal, 2011, 38(10): 12521 - 12530.

[14] OLSTON C, PANDEY S. Recrawl scheduling based on information longevity[C]//WWW '08: Proceedings of the 17th International Conference on World Wide Web. New York: ACM, 2008: 437 - 446.

[15] DONTCHEVA M, DRUCKER S, SALESIN D, *et al*. Changes in webpage structure over time, TR2007-04-02[R]. Washington, DC: University of Washington, 2007.