Available online at www.sciencedirect.com

SciVerse ScienceDirect

Procedia Technology 4 (2012) 566 – 572

Procedia Technology

C3IT-2012

A Reputation Based Detection Technique to Cloaked Web Spam

A.Naga Venkata Sunil[a], Anjali Sardana[a]

[a]Department of Electronics and Computer Engineering, IIT Roorkee, Roorkee,247667, India

Abstract

Web spamming techniques are used by spammers in order to boost one's PageRank of the page and show themselves in top of the results. Spamming techniques can be classified as boosting techniques and hiding techniques. Search engines are facing problems mainly due to hiding techniques. Cloaking is a kind of hiding technique which is used to return different pages to the crawler and the user on their request to the cloaked web server. The cloaked web server always return same page to the user but it returns different pages to different crawlers because different search engines use different PageRanking techniques. Dynamic cloaking is a kind cloaking technique in which web servers intermittently send cloaked and non cloaked pages to crawler on its request. There is no method to detect dynamic cloaking to the best of our knowledge. This paper discusses various techniques that exist to detecting cloaking and an exhaustive comparison between the existing techniques is done. This paper also presents an abstract model to detect cloaking based on reputation.

Keywords: cloaking; reputataion; crawler; webspam; PageRank;

1. Introduction

Najork defined "Web spam refers to a host of techniques to subvert the ranking algorithms of web search engines and cause them to rank search results higher than they would otherwise"[2]. Cloaking is a kind of search engine spamming technique, in which different pages are sent to search engine bot and an ordinary browser on their request to cloaked web server. Even if two requests come from two different crawlers at the same time to the cloaked server, different pages are sent to the crawlers because different search engine uses different ranking algorithms. Z. Gyöngyi and H. Garcia-Molina [1] classified cloaking under hiding techniques because, it is hiding original high quality page from the user. Google [7] defined cloaking as "Cloaking refers to the practice of presenting different content or URLs to users and search engines. Serving up different results based on user agent may cause your site to be perceived as deceptive and removed from the Google index". Because of this cloaking the quality of search engine results are declining, since the cloaked server sends high PageRanked page to the crawler and spammed page to the user. So these pages are indexed in top of the results by search engines for user's query.

doi:10.1016/j.protcy.2012.05.090

There are three methods of cloaking which are used by cloakers, to issue different pages on crawler and browser request. They are as follows:

- IP address: Whenever a HTTP request comes to a cloaked site, it will check the list of IP addresses of the search engine crawlers. If there is a hit then it will send a high quality page to the crawler else it will send cloaked page. Now-a-days there are software's or websites that are providing updated IP addresses of search engine crawlers, so that made the work of cloaker's easy.
- User-agent delivery: In this method the crawlers are identified by seeing the user-agent field of the HTTP request. The browser request and the crawler request are differentiated because they have different user-agent headers and crawler may fake this by placing some browser's application.
- Referer HTTP header: The referer header includes the IP address of the referrer of the request. So a cloaker can easily identify who is the referrer of the request. Like user-agent this header can also be faked.

In this paper we present critical review on existing techniques and presented an abstract model to detect cloaking based on reputation of the URL. Reputation of a URL means assessing the trustworthiness of the URL based on various factors like age of the URL, whether the server is using dynamic IP or not, whether the company is in fortune 500 list or not, etc. We will collect top 200 two hundred URL's of the most popular 100 queries of microsoft's bing search engine. Classify each URL in one among the three categories based on the reputation given by the sites like cisco IronPort SenderBase Security Network [9]. By applying different technique to each URL based on its category, classify the URL whether it is cloaked or not. The contributions of the paper are:

(a) It presents different existing techniques to detect cloaking and an exhaustive comparison between then is done,
(b) Presented an abstract model to detect cloaking using reputation as a constraint,
(c) The proposed technique reduces the computations by detecting cloaking at intermediate steps and reduces memory by using efficient data structures.

Section 2 describes the existing techniques to detect cloaked web spam and exhaustive comparison is made among the existing techniques. Section 3 describes about dynamic cloaking, Section 4 discusses a reputation based technique to detect cloaking. Section 5 discusses about the conclusion and future work to done.

2. Detection techniques for cloaked web spam

Cloaking detection techniques use difference in pages obtained between crawler and browser as major constraint. This section describes various techniques to detect syntactic and semantic cloaking.

2.1 Identifying cloaked web servers:

Najork[8] has proposed a technique to identify cloaked web servers in which first object is collected by sending request to the web server by a crawler and second object is collected by sending request to the server from a ordinary browser. A Web server is identified as cloaked server if both the objects do not match. This method is used to detect cloaking by taking minimum number of copies but it falsely identify the non cloaked web servers also as cloaked web servers, because web is dynamic in nature there may be frequently updated sites

2.2 Detecting cloaking using HITS and HOTS data:

B. Wu and B. D. Davison [3] presented three methods to detect syntactic cloaking using two different datasets.

- Term Difference: Three copies of URL named C_1, B_1 and C_2 are collected. "Bag of words" of each copy of URL is collected, which means collecting all the terms that appeared in the page only once no matter how many times it appear in the page. Calculating the difference in terms between both the crawler's copies (NCC) and calculating the difference in terms between browser's copy B_1 and crawler's copy C_1 (NBC), a page is considered as a cloaking page if NBC>NCC exceeds a predetermined threshold value.

- Link Difference: This method is similar to the previous method in which difference in links is considered.

These two techniques do not provide satisfactory results because a high threshold improves precision at the expense of very low recall. They proposed a third method which automatically detects the cloaking using HOTS dataset. In that method they collected four copies of the page named C_1, C_2, B_1 and B_2.
Algorithm to detect cloaking automatically:

In this method "bag of words" approach has been used. The difference between both the crawler's copy (denoted as HCC) and browser's copy (HBB) is calculated. If the sum of the terms that are present in both crawler's copy but not in browser's copy and terms that are present in both the browser's copy and not in crawler's copy exceeds some threshold value then it is considered as cloaking.

2.3 Cloaking score:

Chellapilla and Chickering[5] proposed a method using the concept of normalized term frequency difference(NTFD). Two different query categories namely popularity and monetizability are used. Popularity of a query is defined to be proportional to the number of times it occurred in the query logs and monetizability as the amount of revenue generated by user clicks on sponsored links. In this method browser's copy B_1 and crawler's copy C_1 are collected, then both the copies are checked to find whether have same HTML or not, if so the URL is marked as non-cloaked, if not HTML is converted into simple text and verified whether both copies have same text, if so it is considered as legitimate URL, otherwise copy of browser B_2 and crawler C_2 are downloaded. Then NTFD can be calculated by using equation (1) :

$$NTFD(T_1,T_2)=1-2(|T_1 \cap T_2|/|T_1 \cup T_2|) \quad (1)$$

Where |.| indicates the cardinality set operation, here the set is multiset of terms. The value of NTFD will always lies in [0,1]. Equation (2) gives the cloaking score S of a URL.

$$S=\Delta_D/\Delta_S \quad (2)$$

Where $\Delta_D=\min\{NTFD(B_1,C_1),NTFD(C_2,B_2)\}$, and $\Delta_S=\max\{ NTFD(B_1, B_2),NTFD(C_1,C_2)\}$. If the value of S exceeds some threshold value then it is considered as cloaked. This method performed well for URLs retrieved from monetizable queries. Choosing a threshold value is not an easy task, since it can be in the interval $[0,\infty)$.

2.4 Detecting semantic cloaking:

B. Wu and B. D. Davison introduced a technique to find semantic cloaking. Semantic cloaking refers to differences in meaning between pages which have the effect of deceiving search engine ranking algorithms [4]. This is a two step method in which the first step is filtering step and second step is classifying step. In first step the difference between crawler's copy C_1 and browser's copy B_1 is calculated. If TB_1C_1 is less than some threshold then it is considered as not-cloaked and filtered in this step. In classification step two more copies are downloaded each from crawler's perspective (C_2) and browser's perspective (B_2). From the four copies B_1, B_2, C_1 and C_2 a set of features are extracted. The features extracted may of Content-based features or Link-based features from each copy. Based on these features a classifier is built and used to detect whether the page is semantically cloaked or not. Content-based features include features like response codes and number of terms in HTTP response header and few other attributes. Link-based features includes number of different kinds of links like total links, unique links etc.

2.5 Tag-Based cloaking detection:

Jun-Lin Lin[7] proposed three tag-based techniques to detect cloaking. Every URL copy is considered as a multiset of tags and standard set operations are performed. In this method tags are used to calculate the difference because the web is dynamic in nature, all the legitimate sites have similar kind of structure even if the content is changed, but a cloaked URL presents different structure to both the crawler and the browser. Here the tag difference between two multisets S_1 and S_2 is shown in (3):

$$|(S_1\backslash S_2) \cup (S_2\backslash S_1)|= |(S_1\cup S_2)|-|(S_1 \cap S_2)| \quad (3)$$

Author defined three formulas and named them as TagDiff2, TagDIff3 and TagDiff4. The difference between the tag's of browser's copy and crawler's copy of a URL are calculated as follows:

$$\text{TagDiff2: } |B_1\backslash C_1| + |C_1\backslash B_1| \quad (4)$$

$$\text{TagDiff3: } (|B_1\backslash C_1| + |C_1\backslash B_1|)-(|C_1\backslash C_2| + |C_2\backslash C_1|) \quad (5)$$

$$\text{TagDiff4: } |(B_1\cap B_2)\backslash (C_1\cup C_2)|+|(C_1\cap C_2) \backslash (B_2\cup B_1)| \quad (6)$$

Equations (4), (5) and (6) are used to calculate the tag difference between browser's copy and crawler's copy. Here B_1, B2, C1 and C_2 are taken as multisets of tags of browser's and crawler' copies of the URL. If difference exceeds some threshold then it is marked cloaked URL. The number after TagDiff indicates the number of copies of the URL. The body spammed page can also be marked as legitimate site if it considers only tag structure.

Comparison of detection techniques:

The comparison between cloaking detection techniques are based on number of copies of URL needed, how the data elements are used i.e. whether a URL copy is considered as set or multiset and drawback of each method. Table 1 gives the comparison between various cloaking detection techniques.

Table 1. Comparison among cloaking detection techniques

Method	Copy	Group	Drawback
Identifying cloaked web servers(Najork)	2	Feature vectors	Identifies dynamic page also as cloaked
Link Difference	4	Set	Only fewer cloaking pages can found because link difference is smaller.
Term Difference	4	Set	Term and link difference falsely identifies dynamic pages as cloaked and they need more number of copies.
Algorithm to detect automatically	4	Set	Falsely identifies dynamic pages as cloaked

Cloaking Score	4	Multiset	Choosing threshold for S is not an easy task [7].
Detecting semantic cloaking	4	Set+Multiset	It needs 4 copies if initial step is not satisfied.
TagDiff2	2	Multiset	Fails in the case, if structure of the page is changed.
TagDiff3	3	Multiset	Fails in the case, if structure of the page is changed.
TagDiff4	4	Multiset	Fails in the case, if structure of the page is changed.

3. Dynamic cloaking:

Jun-Lin Lin defined dynamic cloaking as “the practice of cloaking intermittently to make it difficult to identify”[6]. Dynamic cloaker can switch between “Cloak” and “No Cloak” modes. In “Cloak” it will send keyword enriched page to the crawler, in “No Cloak” mode cloaker will send same version to both crawler and browser.

3.1 Event driven cloaking:

In this method the switching between the modes is done based on crawler event. Here crawler event indicates the request from the crawler. A timer is maintained to note the last crawler event.

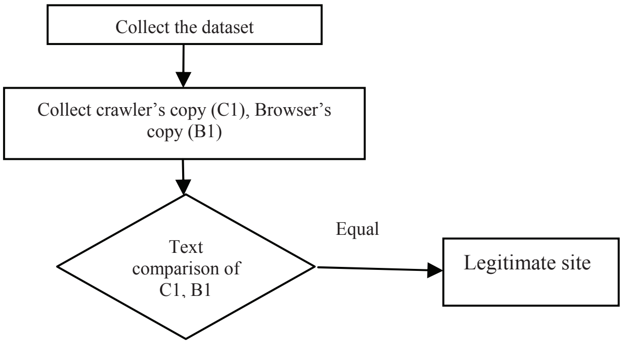
- Initially mode is set to the “Cloak”.
- When a crawler event occurs i.e. when a crawler sends a request, it will check its mode. If it “Cloak” mode it will send different copies to crawler and browser and it will switch itself to the “No Cloak” mode.
- When a time up event occur the cloaker switches to “Cloak” mode, generally time up event will occur if timer exceeds some predefined threshold of last crawler event.

3.2 Monitor table:

This method uses monitor table to store the IP addresses of the crawlers which sends the request and the time of the most recent visit of the crawler. When a crawler event occurs it will check whether the IP address of that crawler is in table or not, if it’s not present then it is appended to the table with the current time and replies the crawler with the high PageRanked page. If the IP address of the crawler is already there then it replies the crawler with non spammed version and it updates the time of that crawler entry with the current time. A time up event occurs if an entry exceeds a predefined threshold.

4. A reputation based technique to detect cloaking

The implementation flow of the proposed technique is given in fig 1. Initially download crawler’s copy (C1) and browser’s copy (B1) of each URL and simple text comparison is made between the copies.


Collect the dataset
Collect crawler’s copy (C1), Browser’s copy (B1)
Text comparison of C1, B1
Equal
Legitimate site

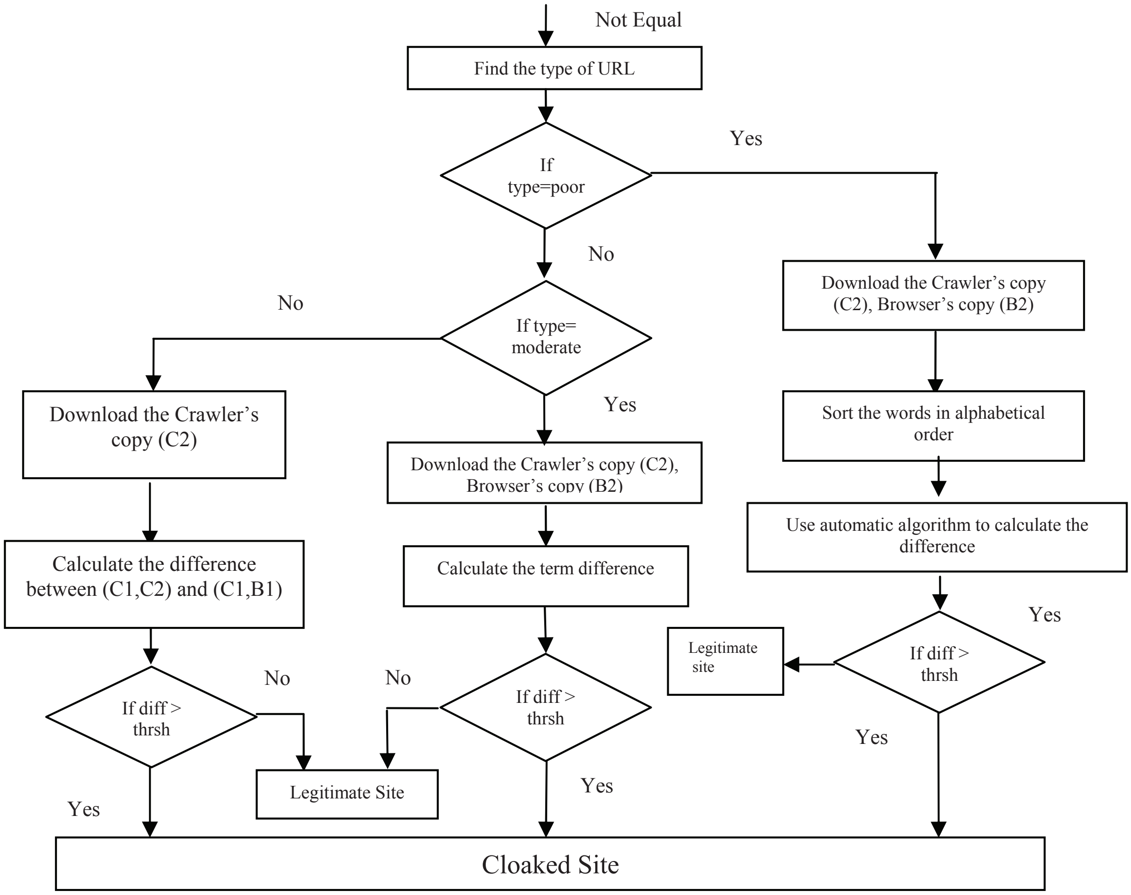

Not Equal
Find the type of URL
If type=poor
Yes
No
Download the Crawler's copy (C2), Browser's copy (B2)
No
If type= moderate
Download the Crawler's copy (C2)
Yes
Sort the words in alphabetical order
Download the Crawler's copy (C2), Browser's copy (B2)
Use automatic algorithm to calculate the difference
Calculate the difference between (C1,C2) and (C1,B1)
Calculate the term difference
Yes
Legitimate site
If diff > thrsh
No
No
If diff > thrsh
If diff > thrsh
Yes
Legitimate Site
Yes
Yes
Cloaked Site

Figure 1: Flowchart of implementation

If they are equal we will declare the page as legitimate page. If they are not equal then classify the URL's into one of the three categories named: good, poor and moderate. Apply cloaking detecting technique starting with pages having high reputation, because of this the pages having poor reputation like spammed pages will never appear in top of the search results. If the URL is a classified under:

- Good: Download another copy of the crawler's copy (C2) and calculate the difference between C1, C2 as NCC and difference between C1, B1 as NCB. If NBC exceeds NCC by some defined threshold (as per experimental results) then classify the URL as cloaked otherwise as legitimate.
- Moderate: Download another copy of the crawler's copy (C2) and browser's copy (B2) and calculate the difference as shown in (6) for words. If the difference exceeds some defined threshold (as per experimental results) then classify the URL as cloaked.
- Poor: Download another copy of the crawler's copy (C2) and browser's copy (B2). Sort the words in pages according to some predefined order. Now calculate the difference by using Equation (6) for each word in decreasing order of their frequencies. If the difference exceeds some predefined threshold value then stop at that instance and declare the page as cloaked. The sorting of the words plays a vital role because the pages with less reputation has high possibility of body spam, redirections, etc. so the difference between the pages sent to the crawler and browser will be more. So they can be identified at intermediate step itself. We will use some efficient data structures to maintain the words according to their frequencies.

So this technique treats each URL differently based on its reputation. If the URL is having high reputation then there is less possibility of cloaking so simply technique is applied. If the URL is having poor reputation there is high possibility of cloaking so we use four copies to detect cloaking using equation (6) on sorted list of words. By evaluating the URL's from high reputation to poor reputation and displaying the results according to evaluation order, the false positives will also never occur in top of the search results. By this way cloaked pages that are not detected by the technique will also never appear in top of the results.

5. Conclusions and future work

This paper proposed a reputation based technique to detect static cloaking, since static cloaking is most common form of cloaking, it is causing severe threat to the search engines. This method has to be practically implemented by collecting large dataset and finding reputation of each URL. The method proposed in this paper uses reputation of the URL to apply appropriate technique on the URL to find cloaking. Most of the techniques have inherent disadvantage of downloading more number of copies and high computations. The proposed technique will reduce the number of computations by detecting at intermediate steps and maintaining the data of the page using efficient data structures.

This paper also presented an exhaustive comparison of existing techniques based on the attributes: number of copies used, group that was used by the method and the drawback of each method. This technique should be examined on real life data set to validate the technique. The technique should be further extended to detect dynamic cloaking to make search engines results free from cloaking.

References

1. Z. Gyöngyi and H. Garcia-Molina. Web spam taxonomy. *In First International Workshop on Adversarial Information Retrieval on the Web (AIRWeb'05),* Chiba, Japan (2005), pp. 1-11.
2. Marc Najork, Web Spam Detection. *In Encyclopedia of database systems, Springer Verlag*, September 2009, Part 23, pp. 3520-3523.
3. B. Wu and B. D. Davison. Cloaking and redirection: A Preliminary Study. *Adversarial Information Retrieval on the Web - AIRWEB* , 2005, pp. 7-16.
4. Baoning Wu and Brian D. Davison. Detecting semantic cloaking on the Web. *15th International World Wide Web Conference, Industrial Track,* Edinburgh, Scotland, May 22-26, 2006, pp. 819-828.
5. Chellapilla, K., & Chickering. D. M. Improving cloaking detection using search query popularity and monetizability. *In Proceedings of the second international workshop on adversarial information retrieval on the web,* Seattle, USA, August 2006,pp. 17-24.
6. Jun-Lin Lin. Detection of cloaked web spam by using tag-based methods. *Expert Systems with Applications journal of Elsevier*, 2009, pp. 7493–7499.
7. Google,http://www.google.com/support/webmasters/bin/answer.py?answer=66355, updated 08/11/2011.
8. M. Najork. System and method for identifying cloaked web servers. Patent Application number 20030131048 (2005).
9. Cisco IronPort SenderBase Security Network. http://www.senderbase.org/index , updated May 2011.