# A Multi-Phase Approach To Retrieve Spam Free Web Pages

**Sreevani. M**

Associate professor, Dept. of CSE,
Mahatma Gandhi Institute of Technology,
Hyderabad, Andhra Pradesh, India sreevani@mgit.ac.in

**Bhramaramba. R**

Associate professor, Dept. of IT,
GITAM University,
Vishakhapatnam, Andhra Pradesh, India
bhramarambaravi@gmail.com

**Vasumati. D**

Professor, JNTUH,
Hyderabad, Andhra Pradesh, India
rochan44@gmail.com

**Rajashree Sutrawe**

Associate Professor, Dept. of CSE,
Arjun College of Technology & Sciences,
Hyderabad, Andhra Pradesh, India
raj.sutrawe@gmail.com

**Yaswanth Babu. O**

IT Manager, TATA Enterprise,
Hyderabad, Andhra Pradesh, India
oyaswanth@gmail.com

## Abstract

Spam web pages have become a major problem in WWW causing both data and network issues. They can corrupt search engine indexes, overload web crawlers and break down web mining services affecting the total performance of web processes. To fix this, we have now proposed a multi-phase analytical approach which recognizes the spam free web pages initially for applying the learned knowledge in later stages to improve the quality of search results and performance of search engines. In the first phase, the spam free web pages will be retrieved using HTTP session information (i.e hosting IP addresses and HTTP session headers). In second phase, a customized page segmentation method will integrate both semantic and fixed length properties of web pages to partition the retrieved pages into blocks. In later phases, the devised algorithms will take advantage of block-level evidence and actualized learning's to improve spam free retrieval performance iteratively and incrementally in specified web context to improve web efficiency by detecting web spam pages before the actual content transfer. Our experimental results proved that effective semantic partitioning of web pages has great potential to improve the performance of current web search engines, effectiveness of search results and utilization of network resources.

## 1 Introduction

Spam free web pages provide good quality and security to web browsers. The malicious Web pages have negative influence on human users and demote the efficiency of Search engines. Thus, for both performance and efficiency reasons, presentation of Spam free web pages has become increasingly important. Many kinds of IR methods have been discovered, but there is no universal method that can retrieve spam free web pages at the same time. Most previous and current research efforts on Web spam defenses have focused on protecting human users by identifying Web pages that skew search engine rankings through link or content manipulations. Representative examples of this research include link-based (L. Becchetti et al, 2006; C. Castillo et al, 2006; C. Castillo et al, 2007; J. Caverlee, 2007; L. Liu, 2007) and content-based (D. Fetterly, 2004; M. Manasse, 2004; M. Najork, 2004; A. Ntoulas et al, 2006; S. Webb, 2007; J. Caverlee, 2007; C. Pu, 2007) analysis techniques. Although these analysis techniques effectively identify certain types of Web spam, the techniques need to combine with IR techniques for good quality web pages retrieval.

We believe HTTP session information (i.e., hosting IP addresses and HTTP session headers) provides sufficient evidence for the successful identification of many Web spam pages. This is supported by (S. Webb, 2007; J. Caverlee, 2007; C. Pu, 2007) which shows that very distinct patterns emerge within the HTTP session information associated with Web spam pages. In this paper, we leverage that observation by using HTTP session information to train classification algorithms to distinguish between spam and legitimate Web pages. To improve the performance of search engine, we used the characteristics of web pages for information retrieval, instead of treating a whole web page as a unit of retrieval (D.Cai, 2004; S.Yu, 2004). The major shortcoming of treating a web page as a single semantic unit is that it does not consider multiple topics in a page. For example, if the query terms scatter at various regions with different topics, it could cause low retrieval precision. It can be argued that a web page with a region of high density of matched terms is likely to be more relevant than a web page with matched terms distributed across the entire page even if it has higher overall similarity. On the other hand, a highly relevant region in a web page may be obscured because of low overall relevance of that page. In this paper, we retrieve spam free web pages using HTTP session information. We built an HTTP session classifier based on predictive technique (S.Webb, 2008; James C, 2008; Calton Pu, 2008), and by incorporating this classifier into HTTP retrieval operations, we are able to detect Web spam pages before the actual content transfer and we use Combined Page segmentation method to partition the retrieved pages into blocks. Our approach is particularly useful for search engines because it enables more efficient and effective Web crawlers, which will generate indexes with higher quality content.

The rest of the paper is organized as follows. Section 2 describes the web page retrieval using HTTP session information. Section 3 discusses the web information retrieval using page segmentation. Section 4 presents methodology of our approach. Section 5 presents Experimental results and we conclude the paper in Section 6.

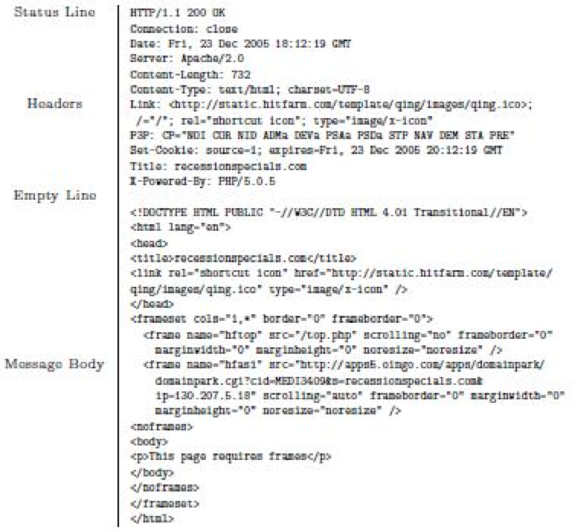
## 2 Web page retrieval using HTTP

To provide the quality of information to web browsers, we use a new approach (S.Webb, 2008; James C, 2008; Calton Pu, 2008) for retrieving Web pages with HTTP. This approach does not affect the underlying HTTP operations. However, it does change the manner in which user agents respond to the results of those operations. Specifically, it inserts an HTTP session classifier into the retrieval process to dramatically reduce the amount of Web spam that is retrieved by user agents. First, the user agent initiates a “GET” request as shown in figure 1. Upon receiving the request message, the origin server performs the requested method and returns the result in a response message as shown in figure 2.

| | |
|---|---|
| Request Line | GET / HTTP/1.1 |
| Headers | Host:click.recessionspecial.com |
| Empty Line | User-Agent:Mozilla/5.0 |

Figure 1: HTTP request message

In the traditional approach, the user agent blindly accepts the response and its corresponding Web page, continuing to its next task i.e., crawling the contents of the next URL, requesting images or other embedded objects that are found in the received Web page, etc.. However, in this approach, the user agent only reads the response line and HTTP session headers from the response message (i.e., it reads up to the empty line). Then, the user agent employs a classifier to evaluate the headers and classify them as spam or legitimate. If the headers are classified as spam, the user agent closes its connection with the origin server, ignoring the remainder of the response message and saving valuable bandwidth and storage resources. Alternatively, if the headers are classified as legitimate, the user agent finishes reading the response message and continues its normal operation. The first advantage of this approach is broadly applicable to any URL on the Web, regardless of where that URL is obtained e.g., in an email, in a search engine result, in a social networking community, etc... Since this approach integrates into the HTTP retrieval process, we are able to protect all of the page requests made by a user agent. Second, by making the classification decision at the HTTP level, we avoid downloading the content of a page unless we predict the page is legitimate. Thus, this approach enables more efficient and effective Web crawlers by providing significant bandwidth and storage savings.


Status Line
Headers
Empty Line
Message Body

```
HTTP/1.1 200 OK
Connection: close
Date: Fri, 23 Dec 2005 18:12:19 GMT
Server: Apache/2.0
Content-Length: 732
Content-Type: text/html; charset=UTF-8
Link: <http://static.hitfarm.com/template/qing/images/qing.ico>;
 /="/"; rel="shortcut icon"; type="image/x-icon"
P3P: CP="NOI COR NID ADMa DEVa PSAa PSDa STP NAV DEM STA PRE"
Set-Cookie: source=1; expires=Fri, 23 Dec 2005 20:12:19 GMT
Title: recessionspecials.com
X-Powered-By: PHP/5.0.5

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html lang="en">
<head>
<title>recessionspecials.com</title>
<link rel="shortcut icon" href="http://static.hitfarm.com/template/
qing/images/qing.ico" type="image/x-icon" />
</head>
<frameset cols="1,*" border="0" frameborder="0">
  <frame name="hftop" src="/top.php" scrolling="no" frameborder="0"
    marginwidth="0" marginheight="0" noresize="noresize" />
  <frame name="hfasi" src="http://apps5.oingo.com/apps/domainpark/
    domainpark.cgi?cid=MEDI3409&s=recessionspecials.com&
    ip=130.207.5.18" scrolling="auto" frameborder="0" marginwidth="0"
    marginheight="0" noresize="noresize" />
<noframes>
<body>
<p>This page requires frames</p>
</body>
</noframes>
</frameset>
</html>
```

Figure 2: HTTP response message.

## 3 Web Information retrieval using Page Segmentation

In this phase we apply Combined Page Segmentation (CombPS) method (D.Cai, 2004; S.Yu, 2004) to partition the retrieved spam free web page into blocks. The combined page segmentation approach called CombPS which tries to take advantage of both visual information and fixed length. The CombPS method is processed as the following two steps:

### 3.1 Step 1. Vision-based Page Segmentation

People view a web page through a web browser and get a 2-D presentation which provides many visual cues to help distinguish different parts of the page, such as lines, blanks, images, colors, etc [22]. For the sake of easy browsing and understanding, a closely packed block within the web page is much likely about a single semantic. We have a vision-based page segmentation method called VIPS (C. Castillo et al, 2007). Similar to semantic passages, the blocks obtained by VIPS are based on the semantic structure of web pages. Traditional semantic passages are obtained based on content analysis which is very slow, difficult and inaccurate. VIPS discards content analysis and produce blocks based on the visual cues of web pages. This method simulates how a user understands web layout structure based on his or her visual perception. The DOM structure and visual information are used iteratively for visual block extraction, visual separator detection and content structure construction. Finally a vision-based content structure can be extracted. Since the method is totally top-down and the permitted degree of coherence can be pre-defined, the whole page segmentation procedure is efficient, flexible and more accurate from semantic perspective.

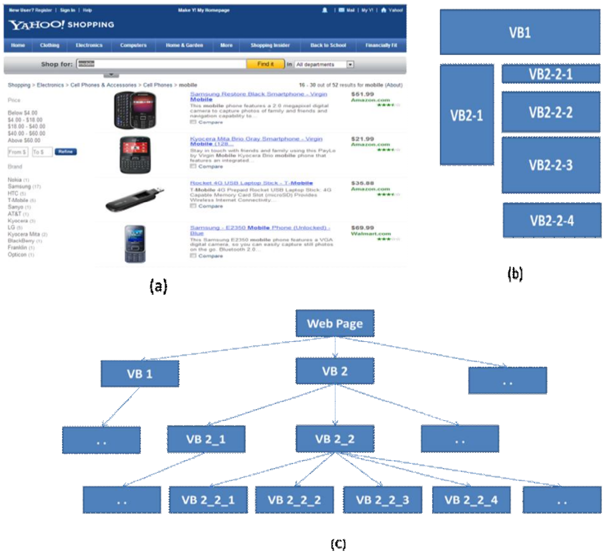


Figure 3: Vision-based content structure for the sample page

In Figure 3, the vision-based content structure of a sample page is illustrated. Visual blocks are detected as shown in Figure 3(b) and the content structure is shown in Figure 3(c). It is an approximate reflection of the semantic structure of the page. In VIPS method, a visual block is actually an aggregation of some DOM nodes. Unlike DOM-based page segmentation, a visual block can contain DOM nodes from different branches in the DOM structure with different granularities. Structural tags such as <TABLE> and <P> can be divided appropriately with the help of visual information, and wrong presentation of DOM structure can be reorganized to a proper form. Therefore, VIPS can achieve a better content structure for the original web page. After the vision-based content structure is obtained, all the leaf visual blocks are taken as the input to the next step for block extraction.

### 3.2 Step 2. Fixed-length Block Extraction

For each visual block obtained in the previous step, overlapped windows are used to divide the block into smaller units. The first window begins from the first word of the visual block, and subsequent windows half-overlap preceding ones till the end of the block. For visual blocks that are smaller than the pre-defined length of the window, they are directly outputted as final blocks without further partition. Upon this strategy, large visual blocks are departed into smaller ones and thus greatly reduce the impact of varying length. Compared with fixed-length approach FixedPS, CombPS utilizes semantic information in partitioning and makes page segmentation insensitive to queries. By allowing small semantic blocks to directly be parts of segmentation results, CombPS intuitively obtains a more diverse and "correct" segmentation result set.

## 4 Web Information retrieval using Page Segmentation

### 4.1 Step 1: Web page retrieval

Initial lists of spam free web pages are retrieved using HTTP session information.

### 4.2 Step 2: Web Information retrieval

A Combined Page segmentation method which integrates both semantic and fixed length properties of web page is applied to partition the retrieved pages into blocks. All of the extracted blocks form a block set. The parameters are set to the same as VIPS. The window length is set to be 200 words.

### 4.3 Step 3: Block Retrieval

This step is similar to Step 1, except that documents are replaced by blocks. The same queries are used to get a block rank BR. After obtaining the block rank, pages can be re-ranked based on the single best-ranked block within each page, though we can also consider several top blocks of each page to re-rank the page.

## 5 Experiments on Multi-phase approach

The success of our multi-phase approach is contingent upon the performance of HTTP session classification. Therefore, we performed extensive evaluations to determine the effectiveness of classifying Web spam using HTTP session information. Afterwards we conducted experiments on block retrieval using the metric precision at 10 (P@10). In this section, we present our experimental HTTP session classification results using spam instances from the Webb Spam Corpus (http://www. webbspamcorpus. org) and legitimate instances from our WebBase (http://www.webbspamcorpus.org) sample. After pre-processing, we get the retrieval baseline of 0.313 for our data. Consequently, the results presented in this section are a truly representative measure of the effectiveness of our approach.

### 5.1 Feature Selection

We expand the feature space with hosting IP addresses and the three feature representations (i.e., phrases, n-grams, and tokens) that we apply to each HTTP header value. To alleviate the problems associated with high dimensionality, we select a smaller number of the "best" features from our vast feature space - a process known as dimensionality reduction (or feature selection). In our experiments, we use a well-known information theoretic measure called Information Gain (Y. Yang, 1997; J. O. Pederson, 1997) to accomplish this feature selection process. Information Gain is defined as follows:

$$IG(f_i, c_j) = \sum_{c\in\{c_j,\bar{c}_j\}} \sum_{f\in\{f_i,\bar{f}_i\}} p(f,c)\cdot log\frac{p(f,c)}{p(f)\cdot p(c)}$$

Where fi is a feature in the feature vector, p(f) is the probability
that f occurs in the training set, cj is one of the classes (i.e., spam or legitimate), p(c) is the probability that c occurs in the training set, and p(f, c) is the joint probability that f and c occur in the training set. Intuitively, Information Gain quantifies how much the knowledge of feature fi helps to correctly identify class ci. Thus, if feature f0 has a higher Information Gain value than feature f1, we say that f0 has more predictive power than f1. For our experiments, we calculate the Information Gain for each feature in the feature space of a given collection of data. Then, a user-specified number n of tokens with the highest Information Gain scores are selected (or retained) and used to train the classifiers.

### 5.2 Classifiers

We performed our HTTP session classification experiments with a variety of classification algorithms. Specifically, we performed an extensive evaluation with number of classifiers that are implemented in the Weka toolkit **[28].** These classifiers include decision trees (e.g., C4.5, random forest, etc.), rule generators (e.g., RIPPER, PART, etc.), boosting algorithms (e.g., AdaBoost, LogitBoost, etc.), logistic regression, radial basis function (RBF) networks, HyperPipes, multilayer perceptrons, support vector machines (SVM), and na¨ıve bayes. Additional information about the classifiers is available in most standard machine learning texts (e.g., T. Mitchell, 1997).

### 5.3 Classifier Evaluation

Our first experiments explored the impact of feature set size and corpus sample size on the effectiveness of our classifiers. As part of these experiments, we varied the feature set size between 100 and 10,000 (incrementing the variations on a log scale), and we varied the corpus sample size across the same range with the same variations. As a result, we evaluated close to 400 unique feature set size and corpus sample size combinations. The 10 most effective features for these corpora are summarized in Table 1.

| Rank | Features |
|---|---|
| 1 | Accept-ranges-bytes |
| 2 | P3p_cp |
| 3 | Server-fedora |
| 4 | Pragma-no-cache |
| 5 | content-type-text/html, charset=utf-8 |
| 6 | content-type-text/html, charset=iso-8859-1 |
| 7 | expires-00 00gmt |
| 8 | 64.225.154.135 |
| 9 | x-powered-by-php/4 3 |
| 10 | x-powered-by-php/4 |

Table 1: Top 10 features for Webb Spam and Web-Base

The performance metrics for the 5 most effective classifiers from our evaluation are presented in Table 2. The table clearly shows that all of our classifiers were quite successful; however, HyperPipes was consistently the best performer.

| Classifier | TP | FP | F-Measure | Accuracy |
|---|---|---|---|---|
| SVM | 89.4% | 2.3% | 0.993 | 93.6% |
| C4.5 | 88.5% | 4.6% | 0.916 | 91.9% |
| Hyper Pipes | 88.2% | 0.4% | 0.935 | 93.9% |
| Logistic Regression | 88.2% | 2.0% | 0.927 | 93.1% |
| RBF Network | 87.1% | 0.8% | 0.927 | 93.2% |
| Content-based | 86.2% | 1.3% | 0.866 | 92.5% |

Table 2 :Classifier performance results for Webb Spam and WebBase.

### 5.4 Block Retrieval

Block retrieval performs the retrieval task at the block level and aims to adjust the rank of documents with the blocks they contain. Through this experiment, our main purpose is to verify whether page segmentation techniques are helpful to deal with both the length normalization and multiple-topic problems. Table 3 shows the experimental results on block retrieval using different page segmentation methods. FullDoc is not listed here since it will always get the baseline. The third column shows the results of using single-best block rank.

| Page Segmentation | Baseline | Block Rank |
|---|---|---|
| DomPs | 0.313 | 0.253 |
| FixedPS | | 0.305 |
| VIPS | | 0.317 |
| CombPS | | 0.327 |

Table 3: P@10 Comparison on block retrieval

For a summarization for block retrieval, DomPS is always the worst and most unstable method, partly because the produced blocks are too detailed and usually cannot be mapped to a single semantic part within the pages. FixedPS gives way to VIPS and CombPS when P@10 is the main concern, partly because it lacks semantic partition and fails to recognize best semantic blocks. VIPS is very good for both data sets in P@10, which means semantic partition is of great importance to web context, CombPS is the best or very close to the best method. This shows that a combination of semantic structure and length normalization is the best choice for block retrieval.

## 6 Conclusion

In this paper, we present a Multi-layered approach to retrieve spam free web pages. Our approach retrieves the spam free web pages using HTTP session information. We used ComPS page segmentation to enhance web information retrieval. Our Experimental results are also very promising. Using our data sets of almost 350,000 Web spam instances and almost 400,000 legitimate instances, our HTTP session classifier effectively detected 88.2% of the Web spam pages with a false positive rate of only 0.4%. We evaluated the effectiveness of these page segmentations for block-level retrieval, and verified that ComPS page segmentation can significantly improve the retrieval performance by dealing with the multiple-topic and mixed-length problems of web pages. We believe such a block-level analysis of web pages will have the opportunity to significantly enhance the performance of existing commercial search engines. We plan to apply this technique to a data set close to the web scale in the future.

## References

L. Becchetti et al. 2006. *Link-based characterization and detection of web spam. In Proc. of AIRWeb.*

A. A. Benczur et al. 2005. *Spamrank - fully automatic link spam detection. In Proc. of AIRWeb.*

C. Castillo et al. 2006. *A reference collection for web spam. SIGIR Forum, 40(2).*

C. Castillo et al. 2007. *Know your neighbors: Web spam detection using the web topology. In Proc. of SIGIR.*

J. Caverlee and L. Liu. 2007. *Countering web spam with credibility-based link analysis. In Proc. of PODC.*

D. Fetterly, M. Manasse, and M. Najork. 2004. *Spam, damn spam, and statistics: Using statistical analysis to locate spam web pages. In Proc. of WebDB.*

A. Ntoulas et al. 2006. *Detecting spam web pages through content analysis. In Proc. of WWW.*

S Webb, J Caverlee and C Pu. 2007. *Characterizing web spam using content and http session analysis. In Proc. of CEAS.*

D Cai, S Yu. 2004. *Block-based web search in proceedings of SIGIR.*

S.Webb, James C, Calton Pu. 2008. *Predicting Web Spam with HTTP Session Information, In proceedings of CIKM.*

D. Cai, S Yu, J -R Wen and W-Y Ma. 2003. *VIPS: a visionbased page segmentation algorithm, Microsoft Technical Report, MSR-TR-2003-79.*

Bailey P, Craswell N, and Hawking D. 2001. *Engineering a multi-purpose test collection for Web retrieval experiments, Information Processing and Management.*

Hearst, M. A. 1994. *Multi-Paragraph Segmentation of Expository Text, In Proceedings of the 32nd Annual Meeting of the Association for Computational Linguistics, New Mexico State University, Las Cruces, New Mexico*:9-16.

Yang, Y. and Zhang, H., 2001. *HTML Page Analysis Based on Visual Cues, In 6th International Conference on Document Analysis and Recognition (ICDAR 2001), Seattle, Washington, USA.*

Bailey P, Craswell N, and Hawking D. 2001. *Engineering a multi-purpose test collection for Web retrieval experiments,Information Processing and Management.*

Y. Yang and J. O. Pederson. 1997. *A comparative study of feature selection in text categorization. In Proc. of ICML.*

*The Web Spam Corpus can be found at http://www.webbspamcorpus.org/.*

T. Mitchell. 1997. *Machine Learning. McGraw Hill.*