# WebGuard: Web Based Adult Content Detection and Filtering System

Mohamed Hammami
*Ecole Centrale de Lyon, France*
*mohamed.hammami@ec-lyon.fr*

Youssef Chahir
*Université de Caen*
*chahir@info.unicaen.fr*

Liming Chen
*Ecole Centrale de Lyon, France*
*liming.chen@ec-lyon.fr*

## Abstract

*This paper describes a Web filtering system "WebGuard", which aims to automatically detect and filter adult content on the Web. WebGuard uses Web crawler to extract relevant data from the Web, combines the textual content, the image content, and the URL name of a Web page to construct its feature vector. WebGuard uses data mining techniques to classify URLs into two classes: suspect URLs and normal URLs. The suspect URLs are stored in a database, which is constantly and automatically updated in order to reflect the highly dynamic evolution of the Web. When working, WebGuard simply captures a user's URL, matches it with the suspect URLs stored in the database and takes an appropriate action - filtering or blocking - according to the result of the analysis. Our preliminary results show that it can detect and filter adult content effectively.*

## 1. Introduction

As a large, widely distributed, global information center, the World Wide Web is growing ever more rapidly. More and richer information sources and services, such as news, advertisements, consumer information and adult content are available on the Web everyday. Simultaneously, user communities are becoming increasing diverse. The openness of the Web allows any user to access almost any type of information. However, some information, such as adult content, is not appropriate for all users, notably children.

Some companies have been researching solutions to this problem. Their products have concentrated on IP-based filtering, and their classification of Web sites is mostly manual. But, as we know, the Web is a highly dynamic information source. Not only do many Web sites appear everyday while others disappear, but also site content (include linkage information) is updated frequently. Thus, manual classification and filtering systems are largely impractical. The highly dynamic character of the Web calls for new techniques designed to classify and filter Web sites and URLs automatically.

In this paper, we propose an adult content detection and filtering system that extends adult content detection accuracy by the usage of both image signature and textual clues for adult web sites filtering. Most existing systems rely, by contrast, only on an analysis of textual information. They consequently fail to detect adult content incorporated inside images.

## 2. WebGuard's Architecture

We built WebGuard over a client-server architecture, shown in Figure 1. Users access WebGuard over the Internet through a Java client applet. The query processor communicates with a feature database and the knowledge base, where the system stores semantic and fact-based metadata, respectively.

The web filter system (WebGuard) aims to block those sites with pornographic or other nudity, and sexually explicit language. It provides Internet content filtering solutions and Internet blocking of pornography, adult material, and many more categories. The Internet will thus become more controllable and therefore safer for both adults and children.

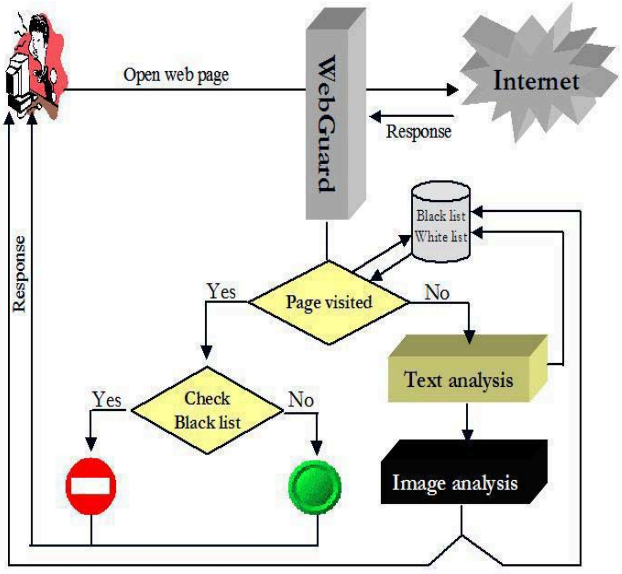


Figure1. WebGuard's architecture

The formulation of the « WebGuard » is as follows:

- Fully automated adult content detection and filtering
- Categorization into "black list" (access denied) and "white list" (access allowed) to speed up navigation
- If the site is not recorded on the "black list" or "white list" the engine will then analyses both the visual and

Proceedings of the IEEE/WIC International Conference on Web Intelligence (WI'03)
0-7695-1932-6/03 $17.00 © 2003 **IEEE**

textual information and makes a further decision on the sites access allowed/denied status. The black list/white list file is then updated.

There is a general consensus regarding certain types of web sites that they must be "filtered" and "blocked" so children do not inadvertently gain access to them. A number of different organizations have created their own definitions of what is or is not appropriate for children using the Internet. The following categories1 are intended to only to serve as a guideline based on the types of materials, text and pictures currently on the Internet:

**Partial Nudity**: These include pictures exposing the female breast or full exposure of either male or female buttocks. This category would not include swimsuits.

**Full Nudity**: Pictures exposing any or all portions of the human genitalia.

**Sexual Acts:** Pictures or text exposing anyone or anything involved in explicit sexual acts and/or lewd and lascivious behavior, including masturbation, copulation, paedophilia, intimacy involving nude or partially nude people in heterosexual, bisexual, lesbian or homosexual encounters. This category would also include phone sex ads, dating services, adult personal ads, CD-ROMs and videos containing the same type of material. Web sites offering the sale of sexual paraphernalia would also be included in this category.

In order to rapidly detect and filter the Web pages with adult sexual content in real-time, we must first have some knowledge about adult sexual content, such as suspected URLs, stored in the knowledge base. Hence, our Web Based Audit Content Detection and Filtering System is comprised of two parts. The first part is designed to create and accurately Update the Knowledge Base (CUKB), the second part is designed to Detect and Filter (D&F) the Web pages with adult sexual content dynamically when younger browsers view them. Figure 2 is the overview of the system architecture.

In CUKB, as show in Figure 3, we have four facilities: the Web Crawler, the Temporary Database, the Data Mining Tools, and the Updating Trigger, used to create and update the Knowledge Base. The Web Crawler is used to periodically search adult sexual images and web pages on the Internet, download suspect images or web pages, put them in the temporary database, and then trigger the Data Mining Tool. The Data Mining Tool uses a data mining method to extract the features of adult sexual images or web pages stored in the temporary database, to discover the suspect URLs, to classify the features, and to trigger Updating Trigger. The Updating Trigger uses predefined strategies to add newly discovered adult sexual content and suspect URLs to the Knowledge Base. To date, we have created the Knowledge Base - and can periodically update it - and have established the fundamentals of our Web based adult content detection and filtering system.

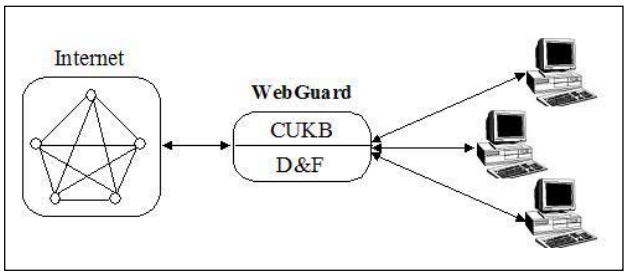


Figure 2. The overview of the system architecture

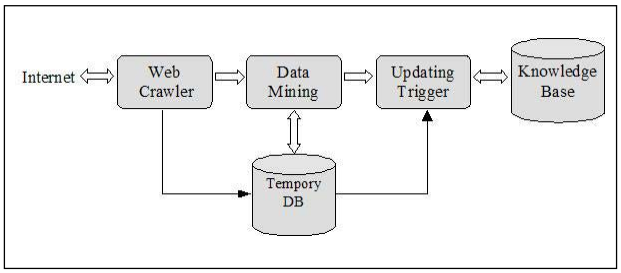


Figure 3. The components of CUKB

In D&F, as shown in Figure 4, we have three facilities to detect and filter browsing activity: the Activity Monitor, the Decision Engine and the Knowledge Base. The Activity Monitor captures active users' URLs in real-time, and compares these URLs with the suspected URLs stored in the Knowledge Base. If such URLs are in the Knowledge Base the Decision Engine is informed. According to the strategies stored in the Knowledge Base, the Decision Engine filters the adult content or disconnects the connection. Apart from classified features and suspected URLs, any anti-browse measures or management information which have been defined by ISPs or generated by the system are also stored in the Knowledge Base.

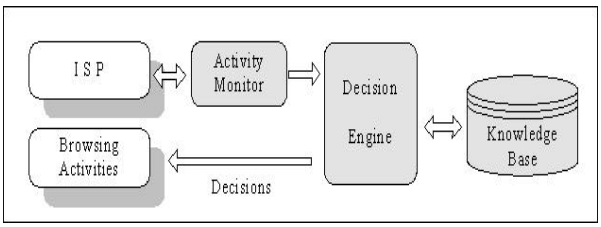


Figure 4. The components of D&F

[1]. Grand Lodge of Ancient, Free and Accepted Masons of Missouri. © Copyright 2000.

## 3. Using Web Crawler to extract feature vectors from Web pages

In order to effectively detect and filter the URLs with adult content, we must first know which URLs are relevant to the adult sexual content. In other words, before detecting and filtering the URLs with adult content, we need to know which URLs are sex oriented and which are not. This is quintessentially a problem of URL classification.

In order to sort the URLs into two classifications, sex-oriented and non sex-oriented, we first decide which features of a URL can be used as its defining features. Considering many sex-oriented Web pages have picture galleries with little or no text at all, we use both image signature and textual clues as the features of a URL. At the same time, many sex-oriented URLs have some pop-up windows and if a Web page links to another Web page it is possible this Web page also has sexual content. Consequently, the number of pop-up windows on a Web page and the nature of a Web page's links (sex relevant or not) are also important features of an URL. The URLs of many sex-oriented Web sites contain sexually explicit words which is another a clear indication that the site contains sexual content.

To summarise the above, we give the feature vector of a Web site as following:

$$\overrightarrow{VoW} = [bSEW, nWD, nWDwS, nLNK, nLNKwS, nIMG, nIMGwS, nPW]^T$$

Where, bSEW is the flag of whether or not the current URL contains sexually explicit words, nWD is the number of words on the current Web page, nWDwS is the number of sexually explicit words on the current Web page, nLNK is the number of links on the current Web page, nLNKwS is the number of the current Web page's links with adult sexual content, nIMG is the number of images on the current Web page, nIMGwS is the number of the current Web page's images with adult sexual content, nPW is the number of pop-up windows on the current Web page.

Using the Web crawler we create the feature vector $\overrightarrow{VoW}$ of a URL. From the definition of the $\overrightarrow{VoW}$ we can know that in order to set up the feature vector $\overrightarrow{VoW}$ of a URL, we should first decide whether or not the Web pages that this Web page linked to are sex relevant. So, we must traverse the Web site corresponding to this URL and get the leaf-URL of this Web site, then construct the feature vectors of all leaf-URLs, then construct the feature vectors of their parent URLs, after which we construct the feature vectors of their grandparent-URLs. Finally, we set up the feature vector of the given URL. Obviously, it is a process of computing from bottom to top. And, in this process, we used stack as the data structure.

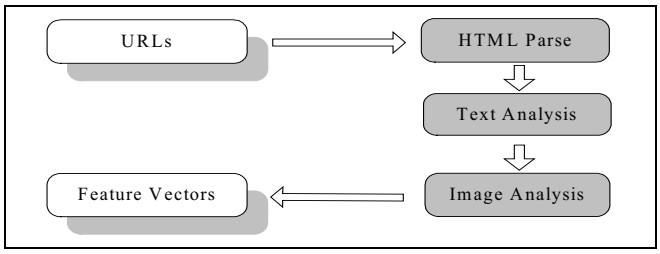


Figure 5. The preparation of feature

As shown in Figure 5, at each step in the computing process, we first parse the HTML, deleting the HTML tags; after that, we analyse the textual content of the HTML, gathering the textual information; and then we analyse the images appearing in the HTML, deciding whether or not they are sex relevant. Finally, based on the obtained information, we create the feature vector of the given URL.

Below are the algorithms used in crawling the Web site and creating the corresponding feature vectors.

**Algorithm 1:** Crawling Web and Computing Feature Vector

```
WebCrawling (url Init_URLs)
For every url in Init_URLs
   all links of the url ← LNKs;
   if url is not leaf-URL and at least one of feature
         vectors of LNKs are not exist
      then { UrlStack.Push(url) ;
         WebCrawling(LNKs) ; }
      else return VectGen( url ) ;
 end for
 while ( ! UrlStack.IsEmpty() )
   UrlStack.Pop-up(url) ;
   VectGen(url);
end while
```

**Algorithm 2:** Computing Feature Vector of given URL:

```
VectGen (url url)
  Check the url to decide whether or not it contains
        the sexually explicit keywords
  Parse the textual content of the url and kick out the
        html tags
  Count the number of words and the number of
        sexually explicit keywords
  Count the number of pop-up windows of this url
  Count the number of sub-urls of this url
  Count the number of images attached to the current
        url, upload and send them to Image Analyzer
  Record all these attributes in Feat_Vect
  Return Feat_Vect
```

## 4. Using Data Mining techniques to classify URLs

Once the feature vectors of all the URLs have been constructed, as shown in Figure 6, the task is to construct a classifier to classify these URLs into two classes: adult sexual URLs and other URLs.

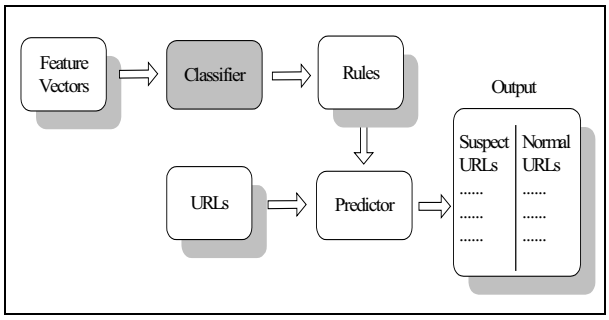


Figure 6. The Classification of URLs

A number of classification techniques from the statistics and machine learning communities have been proposed. A well-accepted method of classification is the induction of decision trees [1, 7]. A decision tree is a flow-chart-like structure consisting of internal nodes, leaf nodes, and branches. Each internal node represents a decision, or test, on a data attribute, and each outgoing branch corresponds to a possible outcome of the test. Each leaf node represents a class. In order to classify an unlabeled data sample, the classifier tests the attribute values of the sample against the decision tree. A path is traced from the root to a leaf node which holds the class predication for that sample. Decision trees can easily be converted into IF THEN rules and used for decision-making.

SIPINA [2] is a widely used technique for data mining. The effectiveness of SIPINA is superior to classical methods such as ID3 and C4.5[8] because these methods often lead to non-desirable situations. These latter instances are caused by deficiencies in their processes, which recognise exclusively divisions, and by insensitive sample sizes. SIPINA tries to reduce the disadvantages of these classical methods by the introduction of the merge operation and a measurement sensitive to the sample size [2]. Let the set of Web Sites be Ω

$$C : \Omega \to \phi = \{\text{ suspect URLs, normal URLs}\} \quad (1)$$

$$W \to C(w) \quad (2)$$

The observation of C(w) is not easy; therefore we are looking for mean value φ to describe class C. The process of graph construction is as follows: We begin with a sample of sites, both suspect URLs and normal URLs and look for the particular attribute which will produce the best partition. We repeat the process for each node of the new partitions. The best partitioning is obtained by maximizing the variation of uncertainty $\Im_\lambda$ between the current partition and previous partition. As $I_\lambda(S_i)$ is a measure of entropy for partition $S_i$ and $I_\lambda(S_{i+1})$ is the measure of entropy of the following partition $S_{i+1}$.
The variation of uncertainty is:

$$\Im_\lambda(S_i) = I_\lambda(S_i) - I_\lambda(S_{i-1}) \quad (3)$$

For $I_\lambda(S_i)$ we use the quadratic entropy:

$$I\lambda(Si) = \sum_{j=1}^{K} \frac{n_j}{n}\left( -\sum_{i=1}^{m} \frac{n_{ij}+\lambda}{n_j + m\lambda}\left(1 - \frac{n_{ij}+\lambda}{ni + m\lambda}\right)\right) \quad (4)$$

Where $n_{ij}$ is the number of elements of class I at the node $S_j$ with I $\in$ {Suspect URLs, Normal URLs}; $n_i$ is the total number of elements of the class $i$, $n_i = \sum_{j=1}^{k} n_{ij}$ ; $n_j$ the number of elements of the node $S_j$ $n_j = \sum_{i=1}^{2} n_{ij}$ ; $n$ is the total number of elements, $n = \sum_{i=1}^{2} n_i$ ; $m = 2$ is the number of classes {suspect URLs, normal URLs ).
As $\lambda$ is a variable controlling effectiveness of graph construction. The algorithm stops if no changes in uncertainty occur.

As a result of applying this method to a training set, a hierarchical structure of classifying rules of the type "IF...THEN..." is created.

## 5. Evaluation and comparison results

Our technique is based on analysing both the textual and visual information to efficiently detect and filter adult content on the WWW. However, there are still several Web sites which could escape our vigilance. These use textual information which does not have a direct relationship to the contents.

In such sites several textual indicators do exist but often these are included within images as in the case of web site “www.france2.com”. This makes accurate automatic adult content detection and filtering difficult. In such situations, the analysis requires another phase of text detection and text recognition in an image [5,6].

Most of the existing systems that rely only on URL and textual information can be readily outsmarted by adult content providers. Still, textual information remains the most significant index for fast filtering and will be used as the first stage in the detection and the filtering of contents of adult sites.

When our method of text based analysis is used we are able to filter more than 90% of web sites. To improve the

performance we use image analysis which is based on our skin color pixel model[3]. According to the percentage of skin color pixels in the image we can decide if the image is suspect. This increases accuracy to 95%.

We evaluated our technique using textual analysis only (A) and then textual + visual analysis using data-mining based skin-color model (B) [4]. For the purposes of our experiment we used 1000 web sites which were manually classified into adult and non-adult sets. There were 500 non-adult web sites and 500 adult web sites. Table 1 shows the improvement of WebGuard system by the use of image analysis.

Table 1: Performance of adult web site detection with textual and visual information

| *Method* | *Site identified* | *Site no identified* |
|---|---|---|
| *A* | *792 (0.792)* | *208 (0.208)* |
| *B* | *988 (0.988)* | *12 (0.012)* |

We have also compared the WebGuard with other Web based adult content detection and filtering systems. The comparison chart is shown in Figure 7. The selected systems are Cyber Patrol, Norton Internet Security, Pure Sight, Cyber sitter, Net Nanny, IE (Internet Explorer).

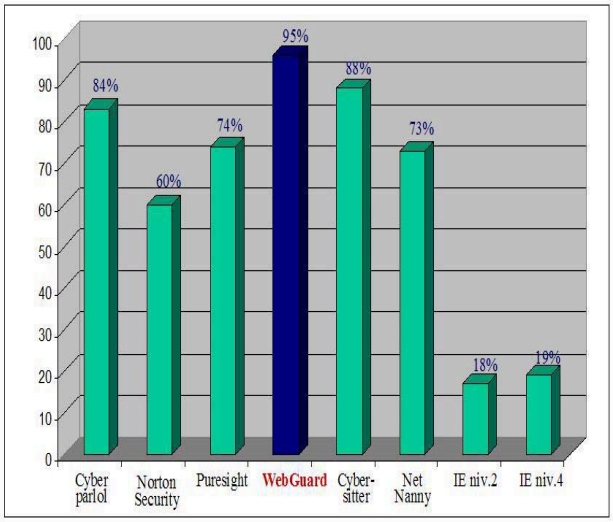


Figure 7. Comparison chart

The comparison has been conducted on 2000 Web sites, including 1000 adult Web sites and 1000 non-adult Web sites. The least effective results come from IE with 18% and 19% success rates while our system is the best with a 95% success rate. Other systems gives success rates between 60% (Norton Security) and 88% (Cyber Nanny).

## 6. Conclusions

In this paper, we have presented a new approach to detecting and filtering Web pages with adult image content in real time. We have proposed WebGuard, which uses data mining techniques to create and update the knowledge base, uses simple matching techniques to detect and filter Web pages, and results in greatly improved system performance.

We have also constructed a robust and effective nudity detector and classifier. This uses a robust skin color model for detecting naked areas in images that are previously suspected.

## Acknowledgement

We thank Pr Qinbao Song (of Department of Computer Science and Technology, Xi'an Jiaotong University P.R. China) for many helpful suggestions.

## References

[1] L. Breiman, J. Friedman, R. Olshen, and C. Stone. *Classification of Regression Trees*. Wadsworth, 1984.

[2] D. Zighed, R. Rakotomala, A method for non arborescent induction graphs. Technical report, Laboratory ERIC, University of Lyon2, 1996.

[3] M. Hammami, L. Chen, D. Zighed, Q. SONG, "Définition d'un modèle de peau et son utilisation pour la classification des images", Ed. Hermès, ISBN 2-7462-0500-9, Juin 2002, pp.186-197.

[4] M. Hammami, Y. Chahir, L. Chen, D. Zighed, "Détection des régions de couleur de peau dans l'image" revue RIA-ECA vol 17, Ed.Hermès, ISBN 2-7462-0631-5, Janvier 2003, pp.219-231.

[5] S. Schüpp, Y. Chahir and A. Elmoataz, Extraction d'informations textuelles dans une vidéo par une approche morphologique robuste , International Conference on Vision Interface VI 2003 , Halifax, Canada Canada

[6] Y. Chahir et al. , "Détection et extraction automatique de texte dans une vidéo: une approche par morphologie mathématique", MediaNet2002, Ed Hermès, pp :73 - 82 , 2002.

[7] J. R. Quinlan. Induction of decision trees. *Machine Learning*, 1:81–106, 1986.

[8] J. R. Quinlan. C4.5: Programs for Machine Learning. Morgan Kaufmann, 1993.