Available online at www.sciencedirect.com
ScienceDirect
Decision Support Systems 42 (2006) 144–161
Decision Support Systems
www.elsevier.com/locate/dsw

A method for managing access to web pages: Filtering by Statistical Classification (FSC) applied to text

Jonathan P. Caulkins [a], Wenxuan Ding [b], George Duncan [a], Ramayya Krishnan [a,*], Eric Nyberg [c]

[a] *The Heinz School, Carnegie Mellon University, Pittsburgh, PA 15213, United States*
[b] *Department of Information and Decision Sciences, and of Computer Science, University of Illinois, Chicago, IL 60607, United States*
[c] *The School of Computer Science, Carnegie Mellon University, Pittsburgh, PA 15213, United States*

Available online 18 January 2005

Abstract

Various entities (e.g., parents, employers) that provide users (e.g., children, employees) access to web content wish to limit the content accessed through those computers. Available filtering methods are crude in that they too often block "acceptable" content while failing to block "unacceptable" content. This paper presents a general and flexible classification method based on statistical techniques applied to text material, that we call, Filtering by Statistical Classification (FSC). According to each individual entity's expressed opinions about what content in a training data set is or is not acceptable, FSC constructs a customized model to represent each individual entity's preferences. FSC then uses this customized model to examine new web content and to block unwanted content. The empirical results suggest that our method has greater predictive power than do a variety of existing approaches.

Keywords: Content-based filtering; Decision support; Statistical classification techniques

1. Introduction

Those who use electronic technology to access information often do not own, control, or otherwise have responsibility for the means of access, nor do they bear the full consequence of its use. For example, employees may use an employer's computer, children may use the family's computer, and the public may access the Internet through library computers. The responsible entity may have an obligation or a preference to regulate what information the user accesses through its facilitation. Prototypical applications are parents protecting their children from age-inappropriate content (e.g., glorification of drug use,

* Corresponding author. Tel.: +1 412 268 2174; fax: +1 412 268 7036. The order of authorship is alphabetical.
E-mail addresses: caulkins@andrew.cmu.edu (J.P. Caulkins), wxding@uic.edu (W. Ding), gd17@andrew.cmu.edu (G. Duncan), rk2x@andrew.cmu.edu (R. Krishnan), ehn@cs.cmu.edu (E. Nyberg).

0167-9236/$ - see front matter © 2004 Elsevier B.V. All rights reserved.
doi:10.1016/j.dss.2004.11.015

instructions on creating bombs, etc.), employers concerned that users might create a hostile work environment for colleagues by accessing inappropriate material such as pornography, and employers preventing employees from spending work time on non-work related content (comics, stock quotes, sports scores, etc.). In the non-electronic context, such filtering is usually non-controversial. A good parent selects age-appropriate books for their children. Advised of a potential for legal liability, employers prevent employees from displaying pornographic magazines and posters in the workplace. Managers seek to keep employees working, not shirking by reading a newspaper's sports pages.

Our goal is to help these responsible entities–call them *guardians*–exercise their legitimate authority to control in the electronic domain what they already control in the physical domain. In an informational context, this means providing an effective filter that provides full and ready access to appropriate material, while blocking access to inappropriate material. This is not an easy task. The variety and extent of material available to the user are so vast, and changing so rapidly, that the guardian cannot possibly catalog and separate the objectionable material from the acceptable. The challenge is to block access to material that a guardian would deem objectionable, even when it is not feasible for a guardian to assess directly the suitability of each site. Conceptually, we address this problem as a classification task. Based on available information, should a site be displayed or withheld? Other categories of action are possible, such as the intermediate one of sending the content to the guardian for evaluation.

We use the specific domain of filtering pornographic content on the web to illustrate our method, because child access to Internet pornography is considered a serious problem [43] and because the inherently visual aspect of much pornography makes it a particularly challenging application (general information about family control of children's access to the Internet is available on the website, GetNetWise (http://www.getnetwise.org/) which as of August 2004, listed more than 65 different software tools for filtering sexual material.) Our method is applied to the text of the subject site, rather than its visual elements. Others are attempting to solve the specific problem of identifying pornographic visual content ([45], also, http://www-db.stanford.edu/pub/gio/2001/wipe-forum.ppt). These efforts are more complementary than competing for at least two reasons. First, they apply only to visual pornography, whereas text-based methods are equally suitable for screening pornographic stories, hate group tracts (for children), sports pages (for office workers), and other classes of material deemed inappropriate. Second, a combination of both approaches is likely to be more effective than either in isolation. Image-based methods may fail when the individuals are partially clothed, and the text-based methods may fail if there are few words on the page, but the chance of both methods failing is lower.

Current methods for filtering based on text include combinations of (1) blacklisting "bad" or whitelisting "good" sites, (2) blocking sites that include "bad" words, and (3) using a "rating" of the web site, whether given by the site creator or a third party, (e.g., the Platform for Internet Content Selection or PICS approach http://www.w3.org/PICS, [36]). These methods have limitations. In the dynamic environment of the web, Method (1) is inadequate because no list of "bad" or "good" sites can be current. Likewise, Method (3) cannot be comprehensive because only a small percentage of web sites can be rated. Furthermore, different people (or guardians) may have different opinions on whether a given web page should be blocked or not, and those differences may not be reflected fully in any simple scale. Method (2), keyword-based filters, breaks down because many "good" sites include "bad" words. Some filtering software uses elaborate context rules, for example, "breast" is bad, unless it appears as "breast cancer," but again, the appropriate context rules can vary by application and even by guardian. For example, Lee et al. [25] use neural networks to filter pornographic web pages. However it does not attempt to customize filtering to fit the value judgments of the guardian. Hence, there is a need for a flexible method that can adapt to each guardian's preferences.

Hence, we have developed a new method, that we call Filtering by Statistical Classification (FSC), to enable personalized control in managing access to web sites. The method uses statistical classification tools through an analysis of certain key features of the subject material in a training data set. The method is (1) *comprehensive*, in that it applies to *all* sites, (2) *customizable*, to reflect the values of a

particular "guardian," and (3) in our demonstration, application *specific* in its capacity to block unwanted sites without blindly blocking all sites that contain suspect words.

The rest of the paper is organized as follows. In Section 2, we briefly review the related research on content filtering. Section 3 describes the FSC method and its customization to a particular application, blocking pornographic web sites. The empirical performance of this application is described in Section 4. A discussion of the FSC method is given in Section 5.

2. Related research in content filtering

At present, there are two basic types of mechanisms for Internet information control. One is blocking software used in conjunction with a conventional search engine. An example is Net Nanny (http://www.netnanny.com), which bills itself as "the world's leading parental control software" and as of January 2004 is in version 5.). The other mechanism is a filtered search engine, such as Family Filter (AltaVista http://www.altavista.com/sites/help/search/family_help), NetWatch (Netscape http://wp.netscape.com/comprod/products/communicator/netwatch/), or Content Advisor (Internet Explorer http://www.microsoft.com/windows/ie/evaluation/features/indepth/contentadv.asp), etc. that processes metadata about each web site based on various systems of PICS labels.

2.1. Blocking software with conventional search engines

In this category, each web browser has a conventional search engine with separate blocking software installed by the computer owner. Of the three methods discussed in Section 1, blocking software use: the *blacklist approach* (Method 1) and the *word filter approach* (Method 2). In the *blacklist approach*, a yes-list or a no-list of URLs is maintained by the service provider to pass or block web sites. When a user submits a search term, if the URL of the search result is in the blacklist, the blocking software will block the corresponding site. Net Nanny is an example of this kind of blocking software. A limitation is the incompleteness of the blacklist: an unacceptable web page that does not appear on the blacklist will not be blocked. Furthermore, sites that may be considered acceptable to the guardian may be in the blacklist. In the *word filter approach*, a list of keywords is maintained by the service provider. This approach works by attempting to match keywords to the user's query term, URL address, or the summary description of the searched web site (stored in both the content of a meta tag of the searched web page and the database of each conventional search engine). In the event of a match, the search is blocked. Since the matching process is limited to attributes such as the URL or meta tags, the content of the page is not parsed or processed and does not play a role in the matching process. It blocks any search whose query term or URL address contains any words in the keywords list and blocks any web page whose meta tag or summary description stored in the database contains any words in the keywords list. From this point of view, word filters do not parse the content of each page. Some blocking software, for example Net Nanny, uses a combination of both the blacklist and word filter methods to block web pages.

2.2. Filtered search engines

In contrast, a filtered search engine has a built-in filter function. There are two types of filtered search engines. The first is a search engine that has a proprietary rating database which contains URL ratings produced by human beings or machines. An example is Family Filter in AltaVista. The other type is a search engine that processes metadata in the form of PICS labels, such as Internet Explorer with its content advisor system. The Platform for Internet Content Selection (PICS), introduced by the World Wide Web Consortium [7,37], stipulates that web pages will have labels that describe relevant aspects of their content, just as the G, PG, R, NC-17, and X rating system does for movies. Then the filtered search engine or the PICS-based web filtering software can be applied to check the label of each web page to determine whether that web page should be blocked. In other words, a parent sets acceptable values for labels that can then be used by the filtered search engine to block or permit viewing of web pages. Hence, the set values or thresholds of the target web page determine which web pages are blocked. Although PICS-based filtering is presumably more

reliable and less error-prone than blacklist and word filtering approaches, guardians are still dependant on the rater's judgment about the site [19,24,30,36,46]. Furthermore, very few sites have PICS ratings and the default behavior of content advisor is to block sites that do not have PICS ratings. This leads to an unacceptable browsing experience.

2.3. Related work in information filtering

Currently there exist two common filtering approaches: individual cognitive filtering [32] and collaborative filtering, a technique used in recommendation systems [2,17,35]. The two approaches differ in the methods used for constructing user profiles and techniques used to calculate relevance of incoming data items.

In individual cognitive filtering, the user's profile consists of a list (vector) of weighted keywords that represent his or her areas of interest. Each item in the vector provides a weight, say on a 0–100 scale, on a particular keyword, expressing the degree of the user's interest in that keyword. In collaborative filtering, the user's profile not only contains a vector of its own keyword weights but also includes a set of sociological attributes such as the user's education, occupation, experience, and a "belong to" relation that indicates collaborative interrelationships between individuals in a community [40]. For a given user, a group of similar users is found based on feedback, recommendations, or individual cognitive profile. Collaborative filtering ranks a document based on a comparison of the user profile to corresponding profiles of the "similar users" in the same community.

Two types of techniques are commonly used in both individual and collaborative filtering: distance and similarity measures, and probabilistic methods [20,44]. Distance and similarity measures, such as the vector space model [4], K-means [16,47], hierarchical clustering [9,18,31], nearest-neighbor clustering [13,20], and latent semantic analysis [12,22], use a selected set of words appearing in different documents as features. Each document is represented as a feature vector, and may be weighted [38]. Then, filtering is accomplished by calculating either the statistical correlation or a cosine value between the vector of keywords that represent the user profile and the vector of features that represent the document to be filtered. The cosine between any two vectors provides a measure of their similarity. If the cosine value is high (or the distance is short), then the document to be filtered is similar to that user's profile (and the document is considered likely to be of interest to the user). Distance and similarity measures can be thought as advanced keyword matching. However, this measure may break down if the size of the document space is large [3].

Another technique is a probability method such as Bayesian classification [5,6,28,41,42]. This technique determines the probability distribution for the features in the training data and uses Bayes' rule to calculate posterior probabilities given a new input. Similar to the distance and similarity measures, Bayesian classification does not perform well when the size of the feature space is much larger than the size of the training set [6]. Also, it generally depends on the conditional independence of the underlying features as well as the model setting used to calculate posterior probabilities. Much of the research on applying these techniques has focused on filtering well-structured documents such as USENET news articles, e-mails, or technical reports [4,8,23,26,27,29,33]. Structured documents make it easier to establish filtering systems because their contents are usually well-defined. However, web documents suffer from both high dimensionality (e.g., large number of features and feature values that are not binary but continuous) and high correlation among feature values [6]. Sarkar and Sriram [39] found the size of the state space increases at an exponential rate when a large number of features are considered or when each feature has multiple values. This is usually true in the case of classifying web documents. Furthermore, filtering pornographic web pages is more difficult because different people have different classification criteria.

The method proposed in the paper is a simple classification algorithm using Probit regression with counts and proportions of text-based markers. We show that our method can identify pornographic content (which typically includes graphical content) better than major existing systems and with a fair absolute degree of precision, and most importantly it can adapt to each guardian's preferences. This illustrates the power of the basic approach of learning decision criteria from a training set in this type of application.

3. The proposed method

3.1. Overall architecture

The system consists of five major types of components: users, a guardian, a private web policy representative, an administrative monitor, and third-party raters (please see Fig. 1 for details on interactions between the components).

A *User* is an individual such as a child whose access to information the system is intended to regulate. The *Guardian* is the individual, family, school, library, organization, business, or other entity responsible for regulating the users' access to content. The users and guardian are people. The *private web policy representative* (PWPR) is a program that is trained to represent the guardian's preferences by examining each web page that users want to access,

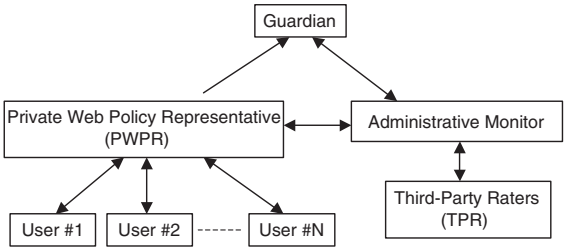


Fig. 1. The system architecture and information flows. Information flow: (1) TPR↔Administrative monitor. A TPR provides training data set(s) and associated assessments or ratings to the Administrative Monitor possibly in exchange for some form of payment. (2) Guardian↔Administrative monitor. The guardian requests customization of the PWPR and subscribes to the service. The monitor creates the custom training data set (that is, the guardian subscribes to an existing third-party rated data set or alternatively may choose to rate the pages in the data set. This is used to create a custom PWPR as noted next. The monitor bills and provides reports to the guardian. (3) Administrative monitor↔PWPR. The administrative monitor combines inputs from the guardian to produce a guardian-specific PWPR who represents the guardian in evaluating web pages. The PWPR reports back a log of activity to the monitor support billing, updating of training data sets provided by the guardian, and a secure repository of information about blocking. (4) PWPR↔Guardian. The PWPR notifies the guardian of filtering activities, e.g., user overrides of soft blocks. (5) User↔PWPR. The PWPR is equipped to filter content requested by users. Each time a user requests a page, the PWPR classifies the page as not-objectionable (in which case the page is displayed), clearly objectionable (in which case the page is hard blocked), or potentially objectionable (in which case the page is soft blocked). The user provides information about pages requested and decision to override soft blocks.

making judgments about the contents of those pages, and taking appropriate action in response to those judgments. Here, the term "page" refers to a block of information that the web policy representative (PWPR) can rate in terms of acceptability and on which action can be taken. It may be an Internet web page or some other chunk of information. When considering an Internet page, it can include both the header and the body of the page, including labels. When the material contains hypertext links, the content pointed to by those links can be taken into account. While in principle, the method applies to a chunk of information (e.g., the content pointed from within a page can be downloaded and added to the chunk of information about which decision has to be made), we do not explicitly address this issue within the scope of this paper.

The term "objectionable" describes content that the guardian does not want the user to access. Guardians may deem content to be objectionable for many reasons, including but not restricted to its being pornographic, violent, racist, not relevant to the users' work tasks, or not relevant to a specific search or request for information. The term objectionable as used in this application can apply to material that is undesirable (e.g., when seeking to block sexually explicit material) or simply not desirable (e.g., when seeking to block information that does not meet specific search criteria).

There are two types of blocking: soft blocking and hard blocking. Hard blocking refers to sites that are simply not displayed. For soft-blocked sites, the PWPR warns the user that the page may violate the guardian's standards. It then gives the user two choices: (1) "Do not display the page" and (2) "Show the page." Thus, the user is given the option to override the soft block and view the questionable page, but if the user does so, that action is reported by the PWPR. One can think of this as "automatic filtering with flexible blocking" rather than "automatic blocking." This flexibility is valuable in application domains where civil liberties strongly protect free access to information (e.g., in public libraries). It is also valuable when the user is both trusted and has time-critical need for access to information. A business, for example, may worry that "hard blocking" might deny critical information to employees working on deadline and/or demoralize employees by treating

them "like children." The soft blocking option could be "turned off" at the guardian's discretion.

A PWPR is trained to represent its owner, a guardian. This training can be done in two ways. One is to use a training set of pages provided by a *Third Party Rater* (TPR) who is a trusted individual or organization other than the guardian. The second option is to use exemplar data generated by the system with the aid of the guardian. This exemplar data represents the guardian's opinion and is used to customize a PWPR.

The default PWPR might be trained using a TPR's data. The system will show the guardian the performance of the default PWPR. If the guardian is satisfied with that performance then no further training is required. If the guardian does not want to use the default PWPR, then the guardian would provide additional training data, either at the outset or by using soft blocked sites.

The last part of the system is an *administrative monitor*, a program resident to those providing the system to the Guardians. It interacts with the guardians, PWPRs, and TPRs. Individual users whose information access is being regulated would only interact with the PWPR. Conversely, third party raters (TPRs) who provide training data sets would interact directly only with the administrative monitor, not the individual PWPRs.

3.2. Customizing a private web policy representative

A page can be blocked because it is deemed objectionable on some combination of criteria, where criteria might include but not restricted to subject matter content (explicit sexual, violent, hate-group, etc.) or types of content (contains pictures, is written in the style of an academic journal, etc.) The rule for aggregating results on the various criteria is called the meta-model. The meta-model could be as simple as, "if the page merits being hard blocked on any criterion, then it should be hard blocked. Otherwise, show it." That simple rule is what we implemented below. Meta-rules could be more complex, also involving interaction among criterion (e.g., block only if sexual content suggests blocking and there is no evidence that it is written in the style of an academic article), the individual user's past history of over-riding soft blocks, or other factors.

To customize a PWPR, the guardian needs to provide preferences about blocking actions for a training data set for each criterion as well as meta-model information. Given this information, the system will (1) identify "markers" based on the training set, and (2) produce guardian-specific model parameters, i.e., a classification scheme. Based on this classification scheme, new pages can be judged to warrant blocking or not.

3.2.1. Identifying markers

Markers can be the presence of objectionable words or phrases (e.g., vulgar words when seeking to screen pornographic material), the presence of reassuring words of phrases (for example, the existence of words such as "statistical significance" or "cancer" may alleviate concerns about the presence of the word "breast" when seeking to block pornographic material), or the presence and nature of pictures or other graphical content. The same object (e.g., word) can represent two markers depending on whether the object would be sent directly to the user or would only be accessible to the user if he or she clicked on a hyperlink emanating from that page (for example, one marker may be appearances of the word "breast" on a given web page; appearances of the word "breast" on pages connected to that page via a hyperlink may be a second, different marker).

Markers can be identified by a variety of processes, either automated or involving human judgment. In our test application (described further below) the training set has 750 web pages (see Appendix B for the determination of the size of the training set). Of these, 375 were "pornographic" pages found by using the first 2 words in a list of the top 100 sex words (see http://www.searchwords.com) that people used most in Internet searches. Typing those two words ("sex" and "porn") individually into the Google search engine produced the 375 pages. The other 375 were not "pornographic." Of those, 114 were "medical" pages obtained by typing "sex education" and "woman health" into Google separately. The remaining 261 were "children's" pages obtained by typing into Google individually 20 search terms a guardian deemed likely to be used by children but having a double entendre. The search terms selected by this guardian (a parent) were kitty, tiger, amateur, doll, bunny, honey, escort, service, doctor, bird, heaven,

beauty, travel, Diana, love, romance, happy, baby, dear, and blouse. We are aware that not all families would agree with this set of terms, but that is one key characteristic of this system; it can be customized to each individual guardian.

We now use this example to show how the system forms markers. For each of the 750 web pages in the training set, the system examines the page's HTML source code, which includes all information that appears in the web page's browser window plus formatting and meta tag information that does not appear in the browser window. The examination consists of six steps [11].

Step 1 For every page, remove all HTML tags, but retain important information, such as the text that appears between the ⟨img⟩ (image) tags, the ⟨content⟩ tags, the ⟨title⟩ tag, and on the hyperlinks.

Step 2 Parse every page in the training set and count the frequency of occurrence of every word. Let P denote the set of words obtained from the 375 pornographic web pages, M denote the set of words obtained from the 114 medical education web pages, and C denote the set of words obtained from the 261 children's web pages.

Step 3 Remove from P, M, and C all prepositions (e.g., in, at, on, over, above, of, off, etc.), articles (e.g., a, an, and the), interrogative terms (e.g., what, how, why, whether, if, which, and that), and words that typically are not relevant to classification problem (e.g., am, are, is, very, will, may, and should).

Some words that appeared in pornographic web pages also appeared in the medical or children's web pages. We need to distinguish between common feature words that appear in all types of pages and those that are specific to pornographic pages. The system uses the following rules (step 4):

Step 4 Let $S=M-P$; $\forall y \in S$, y is in M and not in P. $X=P-\{C \cup S\}$; $C \cup S$ are words in C and/or in S.

Step 5 The 186 words whose frequency is greater than 2 in the set X constitute the "*bad words*" list.

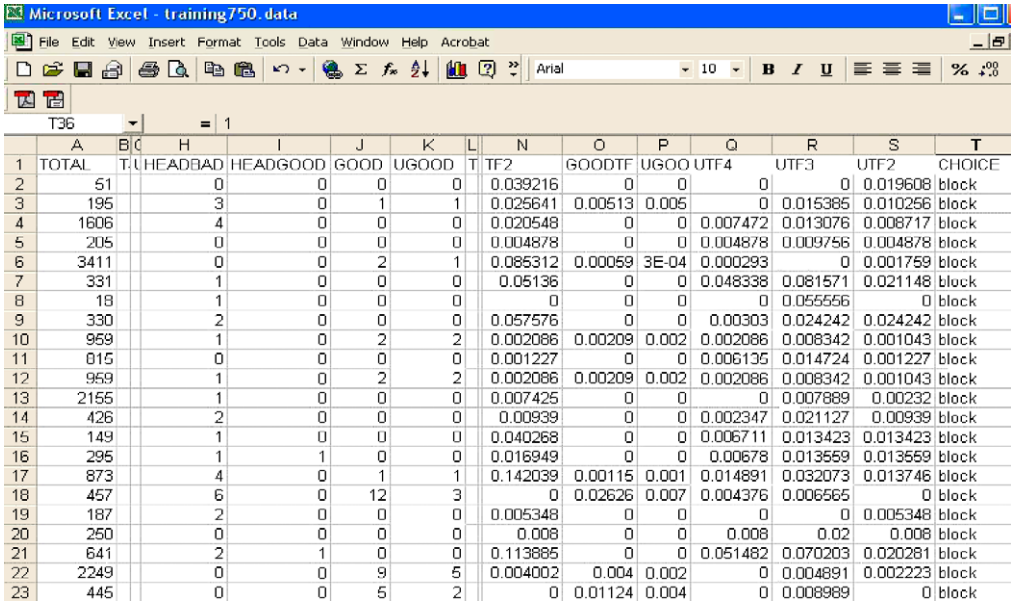
Step 6 The 74 words whose frequency is greater than 2 in the set S constitute the "*good words*" list. We checked these 74 words and confirmed that they did not have any obvious relationship with pornographic content.

The system presumes that repeated occurrence of "bad words" on a page indicates pornographic content. The higher the frequency, the higher the influence that these words have on the property of that web page. Words in the "bad words" list are grouped into 3 levels based on their definitions and the RSACi language rating descriptor (http://www.rsac.org). Level 2, Level 3, and Level 4 indicate that the pornographic content varies from a low to a high degree. Words in Level 4 are strong expletives, words in level 3 are moderate expletives, and words in Level 2 are mild expletives. The presence of "good words" can indicate that a web page is a medical or educational page, such as for science, education, research, pharmacy etc. Therefore, "bad words" in the presence of "good words" may indicate that the content of the page is not pornographic.

These features are aggregated into the following marker variables [11]:

- HEADBAD_n is the number of the "bad words" that appear on the header part of the nth web page;
- HEADGOOD_n is the number of the "good words " that appear on the header part of the nth web page;
- UTF4_n is the term frequency of the unique "bad words" of Level 4 that appear in the body of the nth web page;
- UTF3_n is the term frequency of the unique "bad words" of Level 3 that appear in the body of the nth web page;
- UTF2_n is the term frequency of the unique "bad words" of Level 2 that appear in the body of the nth web page;
- GOODTF_n is the term frequency of the "good words" that appear in the body of the nth web page.

Given these markers, the system produces a classification scheme that can classify a new page. Note that markers such as HEADBAD and HEADGOOD are raw counts while other markers are



Microsoft Excel - training750.data

File Edit View Insert Format Tools Data Window Help Acrobat

T36 = 1

	A	B	H	I	J	K	L	N	O	P	Q	R	S	T
1	TOTAL	T	HEADBAD	HEADGOOD	GOOD	UGOOD	T	TF2	GOODTF	UGOO	UTF4	UTF3	UTF2	CHOICE
2	51		0	0	0	0		0.039216	0	0	0	0	0.019608	block
3	195		3	0	1	1		0.025641	0.00513	0.005	0	0.015385	0.010256	block
4	1606		4	0	0	0		0.020548	0	0	0.007472	0.013076	0.008717	block
5	205		0	0	0	0		0.004878	0	0	0.004878	0.009756	0.004878	block
6	3411		0	0	2	1		0.085312	0.00059	3E-04	0.000293	0	0.001759	block
7	331		1	0	0	0		0.05136	0	0	0.048338	0.081571	0.021148	block
8	18		1	0	0	0		0	0	0	0	0.055556	0	block
9	330		2	0	0	0		0.057576	0	0	0.00303	0.024242	0.024242	block
10	959		1	0	2	2		0.002086	0.00209	0.002	0.002086	0.008342	0.001043	block
11	815		0	0	0	0		0.001227	0	0	0.006135	0.014724	0.001227	block
12	959		1	0	2	2		0.002086	0.00209	0.002	0.002086	0.008342	0.001043	block
13	2155		1	0	0	0		0.007425	0	0	0	0.007889	0.00232	block
14	426		2	0	0	0		0.00939	0	0	0.002347	0.021127	0.00939	block
15	149		1	0	0	0		0.040268	0	0	0.006711	0.013423	0.013423	block
16	295		1	1	0	0		0.016949	0	0	0.00678	0.013559	0.013559	block
17	873		4	0	1	1		0.142039	0.00115	0.001	0.014891	0.032073	0.013746	block
18	457		6	0	12	3		0	0.02626	0.007	0.004376	0.006565	0	block
19	187		2	0	0	0		0.005348	0	0	0	0	0.005348	block
20	250		0	0	0	0		0.008	0	0	0.008	0.02	0.008	block
21	641		2	1	0	0		0.113885	0	0	0.051482	0.070203	0.020281	block
22	2249		0	0	9	5		0.004002	0.004	0.002	0	0.004891	0.002223	block
23	445		0	0	5	2		0	0.01124	0.004	0	0.008989	0	block

Fig. 2. A fragment of a sample data set.

expressed as percentages (or proportions). We use raw counts instead of proportions for HEADBAD and HEADGOOD, since very few words appear in the header and the use of proportions might result in loss of information. On the other hand, in the body, there is usually lots of text, and the use of proportions permits the degree of importance of terms to be captured. A sample data set generated through the process is shown in Fig. 2.

3.2.2. Producing a classification scheme

The system builds a function of these markers and uses the predicted value (Fvalue) to determine if a page should be blocked.

$$\begin{aligned}\text{Fvalue}_n = {} & \beta_0 + \beta_1 \text{HEADBAD}_n + \beta_2 \text{HEADGOOD}_n \\ & + \beta_3 \text{UTF4}_n + \beta_4 \text{UTF3}_n + \beta_5 \text{UTF2}_n \\ & + \beta_6 \text{GOODTF}_n + \beta_7 \text{GOODTF}_n{*}\text{UTF4}_n \\ & + \beta_8 \text{GOODTF}_n{*}\text{UTF3}_n \\ & + \beta_9 \text{GOODTF}_n{*}\text{UTF2}_n + \varepsilon_n \qquad (1)\end{aligned}$$

The dependent variable Fvalue_n is a latent variable which is unobserved by the system. Its values for training data are determined by the model itself during the process of estimation. ε_n is an error term which is assumed to be i.i.d. normally distributed. Note the Eq. (1) includes interactions among markers (independent variables). It is not, in this example, a simple linear function of the markers and as such captures interaction between the proportion of "good words" and "bad words." Note also that the number of unique feature words is counted, not the total frequency of all feature words (repeated words are counted only once).

Values of marker variables may act together to determine categorization decisions. For example, consider some web sites such as medical sites or movie reviews. These sites may contain a few "bad words" but can still be good sites. Based on exploratory examination of the data and findings by Peacefire (www.peacefire.com), we found that this is usually where common misclassification errors happen. Therefore, when capturing the thematic content of such web sites, we need to consider the interaction between the proportion of "good words" and "bad words." So we include three product terms in Eq. (1). Following this logic, eight other nested or non-nested models are examined and shown in Table 5 in Appendix A.[1] We use a Likelihood Ratio Test and the Akaike Information Criterion (AIC) to evaluate

[1] Some other possible models were also considered but did not perform as well as the model in Eq. (1). Due to space limitations, we did not include them in the paper.

Table 1
Effects of independent variables on the predicting probability when choice=1

Marker variables	Amount of variation increase 10%	Amount of variation decrease 10%
HEADBAD	1.33%	−1.85%
HEADGOOD	−0.07%	0.07%
UTF4	0.90%	−0.92%
UTF3	1.28%	−1.75%
UTF2	0.37%	−0.42%
GOODTF	−0.10%	1.51%

the proposed model (Eq. (1)) and other models since these are widely used for comparing different nested and non-nested models [1,14]. The proposed model turned out to have the best overall performance. Appendix A discusses the choice of functional form in detail.

The system uses Eq. (1) to model the relationship between the guardian's choice and the thematic content of a web page under the assumption of normal distribution of the error term [11]. So, we have

$$\begin{aligned}\overline{\text{Fvalue}_n} = {} & \beta_0 + \beta_1 \text{HEADBAD}_n + \beta_2 \text{HEADGOOD}_n \\ & + \beta_3 \text{UTF4}_n + \beta_4 \text{UTF3}_n + \beta_5 \text{UTF2}_n \\ & + \beta_6 \text{GOODTF}_n + \beta_7 \text{GOODTF}_n{*}\text{UTF4}_n \\ & + \beta_8 \text{GOODTF}_n{*}\text{UTF3}_n \\ & + \beta_9 \text{GOODTF}_n{*}\text{UTF2}_n \end{aligned} \quad (2)$$

and employ the following Probit model [14]:

$$\begin{aligned}\text{prob}(\text{CHOICE} = 1) &= \int_{-\infty}^{\overline{\text{Fvalue}_n}} \phi(t)dt \\ &= \Phi\left(\overline{\text{Fvalue}_n}\right) \end{aligned} \quad (3)$$

$$\text{prob}(\text{CHOICE} = 0) = 1 - \text{prob}(\text{CHOICE} = 1) \quad (4)$$

where $\overline{\text{Fvalue}_n}$ is the deterministic part of Fvalue_n, ϕ and Φ denote the probability distribution function and cumulative distribution function of the standard normal distribution, respectively. CHOICE is a binary variable such that CHOICE=1 if the web page is inferred to have pornographic content; otherwise, CHOICE=0.

3.2.3. Sensitivity of model to particular predictors

To see how sensitive the classification scheme is to changes in various markers, we first apply Eqs. (2)–(4) to the training set to get each parameter value (i.e., β_0–β_9). Next we treat all pages in the training data set as one big page. Within this one big page, we calculate an average value for each marker. We then apply those parameter values (β_0–β_9) to Eq. (2) again to get an average value for $\overline{\text{Fvalue}_n}$. Next using Eq. (3), a probability value for this $\overline{\text{Fvalue}_n}$ when choice=1 is obtained and denoted as ave-prob. We then use this ave-prob as a measure to explore the extent of variation in the probability value when each individual marker value is increased or decreased by 10%, holding all other markers constant.

Table 1 displays the results with a row for each marker variable and columns indicating the amount of variation in the marker. The cells give the change in the predicting probability when the marker in the corresponding row has changed by the amount indicated by the corresponding column. For example, in row 2, the marker variable is HEADBAD, so we see that increasing the bad word count by 10% would increase the predicting probability by 1.33%. Likewise, decreasing HEADBAD by 10% would reduce the predicting probability by 1.85%.

Table 1 shows sensitivity to changes in individual markers. The consequences of simultaneously varying all good and/or all bad word counts is shown in Table 2. Overall the two tables show that (1) all markers–except perhaps HEADGOOD–materially influence the evaluation but (2) the system's judgments are directly affected more by systemic variation across multiple markers than by fluctuation in any particular marker.

Table 2
Variation in predicting probability when many markers change simultaneously

Variation in marker variables	Variation in probability (Choice=1)
Increase 10% in good words and decrease 10% in bad words	−11.50%
Increase 20% in good words and decrease 20% in bad words	−35.42%
Increase 50% in good words and decrease 50% in bad words	−93.81%

4. Empirical study

4.1. Test data

Three data sets were used to test the performance of the system.

Test sample 1—This is a data set consisting of 930 web pages collected via web searches using Yahoo. We chose nine search terms (sex, adult, lingerie, cracks, penthouse, swingers, abortion, breast, and legs) from a list of the "top 100 sex words" used in Internet searches (http://www.searchword.com) as well as the terms "gender study" and "women in film." We typed these terms into the Yahoo search engine individually. For every search, we randomly selected one out of every five pages in the list of the search results returned by Yahoo, yielding 930 web pages.

Test sample 2—chat room name data. This is a data set consisting of 222 web pages. We noted the names of all 748 alt. groups at http://www.deja.com/usenet. Some of the names that might attract the attention of children include Barney, hobgoblin, Barbie, booger, cats, clueless, cookies, cyberpunk, Disney, dreams, duck, fantasy, fiction, fun, games, god, horses, mountain bike, movies, mtv, music, penpals, pets, pizza, rap, rave, rock, school, sports, teachers, teens, toys, trucks, tv, videogames, and wedding. We typed these words into the AltaVista search engine (without the family filter option) individually. For each search, we picked up the first 6 or 7 pages in the list of documents produced. This yielded 222 web pages.

Test samples 3a and 3b—Peacefire data. Peacefire developed software to decrypt the database of sites blocked by X-stop (http://www.xstop.com) as obscene. After extracting the first 50 URLs in the ".edu" domain that was current as of January 17, 2000, Peacefire visited those pages and found that 34 of them, or 68%, were obviously errors in X-stop's list. The list of these 50 examined sites is at: http://peacefire.org/censorware/X-stop/xstop-blocked-edu.html. We use those 50 URLs blocked by X-stop to test our method.

Peacefire did the same analysis on I-Gear's blocked-site list that was current as of February 20, 2000. Peacefire examined 50 URLs that were blocked in the "Sex/Acts" category, defined by I-Gear as follows: "Sites depicting or implying sex acts, including pictures of masturbation not categorized under sexual education. Also includes sites selling sexual or adult products." The list of 50 examined sites is at: http://peacefire.org/censorware/I-Gear/igear-blocked-edu.html. We use these 50 URLs blocked by I-Gear to test our method.

For all test data sets, an independent rater, acting in the role of a guardian, supplied his decisions as to whether these web pages should be blocked or not.

Table 3
Comparison of the proposed method to several other methods

Data set	Methods	Sensitivity	Specificity
Test sample 1: 930 pages	PICS	100	0
(808 pages are bad,	Net Nanny	68	88
122 are good)	C4.5	88	84
	Our method	90	86
Test sample 2: chat room	Family Search	100	83
name data (221 pages	C4.5	100	99
are good, 1 is bad)	Our method	100	100
Test samples 3a and 3b:	X-stop	100	0[a]
Peacefire data sets	C4.5	75	94
(X-stop: 16 pages are	Our method	81	100
bad, 34 are good;	I-Gear	100	0[a]
I-Gear: 12 pages are	C4.5	50	91
bad, 38 are good)	Our method	75	95

[a] The samples were picked from .edu domains, so the error rates here do not cover all domains.

4.2. Results

The test samples were used to compare the performance of the proposed system with several popular existing methods or software (such as PICS, Net Nanny, Family search, X-stop, and I-Gear) and with a decision tree technology (C4.5), principally with respect to two performance indices: sensitivity and specificity. For the pages where the guardian considers the material objectionable, the percentage of pages the system rejects is the *sensitivity*. For the pages the guardian considers benign, the percentage of pages the system passes is the *specificity*. The results are summarized in Table 3.[2] Higher sensitivity and specificity indicate a good performance.

[2] Our specificity and sensitivity measures are directly related to the precision and recall measures used to evaluate information retrieval systems [38]; specificity is equivalent to precision, and sensitivity is equivalent to recall.

4.2.1. Test sample 1

For test sample 1, the proposed system was compared first with the PICS approach with NetWatch (http://home.netscape.com/communicator/netwatch/), Netscape Navigator's built-in rating protection feature. NetWatch recognizes two independent PICS-compliant rating systems, RSACi (http://www.rsac.org/) and SafeSurf (http://www.safesurf.org/). We chose to have NetWatch filter pages that have been rated with RSACi or SafeSurf. The NetWatch setup process asks for an "acceptable level." We chose 0 in the RSACi rating and 1 in SafeSurf, which yields the greatest protection. NetWatch compares the page's rating with the levels determined to be acceptable. If a page's levels are higher, NetWatch prevents the page from being displayed. Also, if a web page does not have a label, it is blocked. In test sample 1, only 106 of 930 pages had labels. Because of this and the stringent levels of acceptability selected, NetWatch with PICS blocked all 808 "bad" sites, but also blocked all 122 "good" sites. At least for these data and levels, this is clearly not a very discriminating filter—it has low specificity.

Net Nanny performed much better in this regard. It lets 88% of the good pages be viewed, but it also lets almost one-third of the bad pages through. Our system's type I error rate (1-sensitivity fraction) was only 1/3 as great as Net Nanny's (allowing 1/9 not 1/3 of the bad sites), while blocking only slightly more of the good sites (13.9% vs. 12.3% for Net Nanny).

4.2.2. Test sample 2: chat room name data

We compared our system's performance on the test sample 2 to that of Family Search, a filtered search engine that incorporates content rating (http://www.altavista.com/help/search/family_help). The Family Search service operates as follows. In the AltaVista Search web page, Family Search is turned on by setting Family Filter preferences from the Family Setup page. When the user types a search term into the AltaVista search engine, the AltaVista results are filtered through Net Shepherd's ratings database, and the filtered results are presented to the user.

There is only one bad site (http://www.cybersim.com/booger/default.htm) in this test sample. Unfortunately, this site is no longer available as it was removed on February 12, 2004. Both Family Search and our method blocked it. Hence, both methods have 100% sensitivity. But our system has much higher specificity (100%) compared with the Family Search (83%).

4.2.3. Test samples 3a and 3b: Peacefire data sets (X-stop50 and I-Gear50 data)

As noted before, Peacefire (http://www.peacefire.com) conducted analyses on the blocked lists of X-stop and I-Gear. X-stop has a library (including ethnic and racial slurs, foul words, etc.), an "allowed list," and a "blocked sites list." X-stop updated its blocked sites list daily (old link: http://update.xstop.com/, new one http://www.8e6home.com/kbe.asp?kbid=gen0063). X-stop blocks any search whose query terms are in the X-stop library and any URL that is on the "blocked sites" list, X-stop blocks the site. If the words are used to search for pornography and other objectionable sites on the Internet, X-stop blocks the search. I-Gear (http://www.symantec.com/sabu/igear/index_news.html) has a similar function as X-stop.

Since both Peacefire data sets were taken from lists of blocked sites, X-stop and I-Gear blocked 100% of their respective sites, yielding 100% sensitivity. However, they had 0% specificity. Our system was able to block four-fifths of the X-stop list's bad sites while blocking none of the good sites. Likewise, it blocked three-fourths of the I-Gear list's bad sites, but only 2 of the 38 good sites.

4.2.4. Comparison with decision tree-based filtering using C4.5

C4.5 is a program used widely in Data Mining to find data patterns. It can induce classification rules in the form of decision trees from a set of given examples [34]. We applied C4.5 to our training data so that it could generate some decision rules based on our marker variables. Then we used these generated decision rules to examine our other test samples. The results are also displayed in Table 3. C4.5 did well in terms of sensitivity and specificity with the training data set (sensitivity=92 and specificity=97). It also compares favorably with the other software listed above and even with our method on the easier data sets (test sample 1 and 2), but not as well overall, particularly on the harder Peacefire data sets.

C4.5 essentially utilizes pre-defined keywords to search for (or induce) "rules" from a set of given data

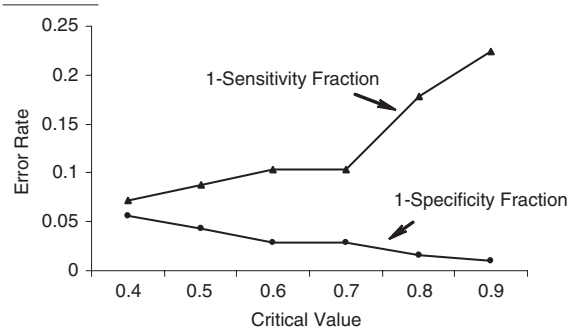


Fig. 3. Sensitivity and specificity in training data as a function of the Probit regression's critical value.

(for example, a web page has been summarized based on the markers). It does not model the underlying data generating process, as we did in our proposed model. To compare its predictive ability with that of our proposed model, we input those keywords or "markers" used in our proposed model into C4.5 and then compute the predicted results. Since the keywords are the same and C4.5 induces the "rules" fitting the training sample, it is not surprising that the prediction results using the training sample for the two models are not statistically significantly different. (The z-test for the difference of sensitivity for the two approaches is -0.7314, p-value=0.2327; the z-test for the difference of specificity for the two approaches is -1.267, and the p-value=0.102.) On the test samples, however, since C4.5 does not model the underlying data generating process and our approach does, overall the predictions of our model are significantly better than those of C4.5 (the z-test for the difference of sensitivity for the two approaches is 2.0081, p-value=0.0228; the z-test for the difference of specificity for the two approaches is 1.6583, and the p-value=0.0485). Because both of the p-values are less than the widely used 5% significance level, both of the null hypotheses of no difference between the two methods are rejected. These results suggest that generating "markers" which capture or summarize the contents of a page is useful and our proposed model has overall better predictive power.

4.3. Summary and comparison using other criterions

Even with the critical value in the Probit regressions arbitrarily set to 0.5, our system was much better at allowing "good" sites to be shown while still blocking the majority of bad sites than were the competitors. Furthermore, our system's performance could be improved and customized to the needs of a particular guardian, by tuning the balance between sensitivity and specificity by adjusting the critical value in the Probit regressions. For example, using the training data set selected by a particular guardian, Fig. 3 shows other combinations of sensitivity and specificity that our system could achieve, just by varying the critical value setting in the Probit regression. This analysis can be used by the guardian to select the critical value for the Probit regression when it is used to classify any new web page.

4.3.1. Comparison on rating cost and time lag

Our proposed method compares favorably to the alternatives on other metrics such as coverage of web pages, cost of hiring human raters, ease of use, and time lag. These are summarized in Table 4. We highlight rating costs and time lag in the following discussion.

We use rating cost to refer to the cost incurred by human raters to create the ratings/information used in filtering pages. In the PICS-based approaches, each page that is to be filtered has to have a PICS rating. Search engines such as Family Search work with a rating database maintained by human raters. The same is true of X-stop which states on its web site (http://www.8e6home.com/kbe.asp?kbid=gen0063) that "We have a team of people who scan the Internet daily using automated tools and manual methods to

Table 4
Comparison of our method to other alternatives

Method	Coverage of web pages	Cost of hiring human	Ease of use	Time lag[a]
PICS	Low	High	Difficult	High
Net Nanny	Middle	Middle	Easy	High
Family Search	Middle	High	Easy	High
X-stop	Middle	High	Easy	Middle
I-Gear	Middle	High	Easy	Middle
Our method	High (100%)	None	Easy	None

[a] Time lag means the time spent on considering whether the target web page should be either "labeled" or put in the "blacklist."

produce a database of blocked sites. The database is maintained at various regional locations, and is accessed by a service component which is downloaded from us and runs on your computer." Thus PICS-based approaches, Family Search and sites like X-top and I-Gear incur high rating costs since their method relies on rating by page. Net Nanny uses a manually updated list of words and the creation of these is not as expensive as rating individual sites and pages. Our approach does not incur these types of rating costs. However, each guardian has to rate a set of pages in the training data set. While this cost to the guardian can be offset by using a third party, it is nonetheless a cost that has to be incurred to achieve customization. Once the model is estimated, it can be used to filter pages at no additional cost to the user or the guardian.

Time lag is the sum of the time that it takes to rate a new page or to label it or to create structures such as blocked lists and the time taken to make this information available for use in filtering. Approaches such as Net Nanny, PICS, and Family Search have a high time lag since human raters are required to read web pages to build structures such as yes-list and no-list and because these new structures are not updated with high frequency for use in filtering. X-stop and I-Gear update use a combination of automated tools and human raters to create their blocked URL list but update their database daily and on account of this their time lag is not as high. Our proposed approach does not incur a time lag once the method's parameters have been estimated since the approach does not rely on evaluating new pages with human raters.

5. Discussion and conclusions

The goal of this paper is to improve on prior methodology for intelligently filtering potentially objectionable content. The proposed method of managing access to pages based on processing content of each page is comprehensive in that it applies to all pages of information. In contrast, labeling schemes such as PICS have nothing to say about pages that have not been labeled. Our method is customizable since the private web policy representative (PWPR) can be customized to reflect the values of a particular guardian simply by adjusting the training set. It is specific in its capacity to filter unwanted sites without blindly blocking all sites that contain suspect words or content because it does not rely on a simple word filter of the sort that has attracted condemnation from organizations such as Peacefire. It does not fall out of date because judgments are based on the actual content of the page at the time of the request. In contrast, labeling schemes, black-lists, and white-lists of URLs can fail if the page content has changed since the label was created or the listing decision was made.

The proposed method is also customer-friendly on several dimensions. First, it can be customized to a guardian's preferences without the guardian necessarily having to understand the filtering algorithm or specify what it is that makes a page objectionable. Although the guardian can request certain markers, the method provides a default set of markers and the guardian need to only indicate which pages in the training set are objectionable and which are not for a given criterion. Simple binary judgments are all that is required of the guardian. A still simpler option for the guardian is adopting the training set provided by a third-party rater.

Second, our system monitors the user's override activities, whereas most existing products can be overridden by users by moving URLs from forbidden-lists to allowed-lists or deleting feature keywords, etc. The proposed method processes the content of each page to form a function of markers that represents the theme of that page. The only interaction between the system and the user is through the PWPR, but the PWPR only has parameters instead of markers. There is no way for the user to know what type of markers the system uses to form the theme function.

Third, the system can adapt even if website producers frequently change their web pages. The system captures the meaning of each web page by processing its contents instead of just the title or text on hyperlinks.

In conclusion, we have proposed and illustrated a simple and general method for classifying a web page as objectionable or not, according to the individual standards of a guardian. The innovative aspects of the proposed method are these:

1) Statistical classification methods, to our knowledge, have not been systematically employed as a

tool in controlling access to pornographic or other objectionable material as it exists on the web.

2) Different "guardians" (who might be parents or employers) have different "values" or opinions about what should or should not be blocked. Thus the ability to customize the filter to the particular guardian's values is essential; the proposed method can do this.
3) Customized training of the filter can be time consuming. Our solution to that problem is to allow guardians to align themselves with one of several profiles based on the opinions of some recognized authorities from whom detailed training information has been elicited.

Appendix A. Examining the validation of the proposed model (Eq. (1))

To examine the validation of the concept model proposed above, we analyze the following nine models. These nine models were selected based on exploratory examination of the web pages in the data and findings by Peacefire (www.peacefire.com) which showed that common misclassification errors happen when "good" web pages such as medical web pages or movie review pages contain a few "bad" words or vice versa. So the interaction terms are examined between markers with "good" words and markers with "bad" words. The nine models are

$$\text{Fvalue}_n = \alpha_0 + \alpha_1 \text{HEADBAD}_n \quad \text{(Model 1)}$$

$$\text{Fvalue}_n = \delta_0 + \delta_1 \text{HEADBAD}_n + \delta_2 \text{HEADGOOD}_n \quad \text{(Model 2)}$$

$$\text{Fvalue}_n = \rho_0 + \rho_1 \text{HEADBAD}_n + \rho_2 \text{HEADGOOD}_n + \rho_3 \text{HEADBAD}_n{*}\text{HEADGOOD}_n \quad \text{(Model 3)}$$

$$\text{Fvalue}_n = \mu_0 + \mu_1 \text{HEADBAD}_n + \mu_2 \text{UTF4}_n + \mu_3 \text{UTF3}_n + \mu_4 \text{UTF2}_n \quad \text{(Model 4)}$$

$$\text{Fvalue}_n = \kappa_0 + \kappa_1 \text{HEADBAD}_n + \kappa_2 \text{HEADGOOD}_n + \kappa_3 \text{UTF4}_n + \kappa_4 \text{UTF3}_n + \kappa_5 \text{UTF2}_n + \kappa_6 \text{GOODTF}_n \quad \text{(Model 5)}$$

$$\text{Fvalue}_n = \eta_0 + \eta_1 \text{HEADBAD}_n + \eta_2 \text{HEADGOOD}_n + \eta_3 \text{UTF4}_n + \eta_4 \text{UTF3}_n + \eta_5 \text{UTF2}_n + \eta_6 \text{GOODTF}_n + \eta_7 \text{GOODTF}_n{*}(\text{UTF4}_n + \text{UTF3}_n + \text{UTF2}_n) \quad \text{(Model 6)}$$

$$\text{Fvalue}_n = \lambda_0 + \lambda_1 \text{HEADBAD}_n + \lambda_2 \text{HEADGOOD}_n + \lambda_3 \text{UTF4}_n + \lambda_4 \text{UTF3}_n + \lambda_5 \text{UTF2}_n + \lambda_6 \text{GOODTF}_n + \lambda_7 \text{GOODTF}_n{*}\text{UTF4}_n + \lambda_8 \text{GOODTF}{*}\text{UTF3}_n + \lambda_9 \text{GOODTF}_n{*}\text{UTF2}_n \quad \text{(Model 7)}$$

$$\text{Fvalue}_n = \gamma_0 + \gamma_1 \text{HEADBAD}_n + \gamma_2 \text{HEADGOOD}_n + \gamma_3 \text{UTF4}_n + \gamma_4 \text{UTF3}_n + \gamma_5 \text{UTF2}_n + \gamma_6 \text{UGOODTF}_n \quad \text{(Model 8)}$$

$$\text{Fvalue}_n = \tau_0 + \tau_1 \text{HEADBAD}_n + \tau_2 \text{HEADGOOD}_n + \tau_3 \text{UTF4}_n + \tau_4 \text{UTF3}_n + \tau_5 \text{UTF2}_n + \tau_6 \text{UGOODTF}_n + \tau_7 \text{UGOODTF}_n{*}\text{UTF4}_n + \tau_8 \text{UGOODTF}_n{*}\text{UTF3}_n + \tau_9 \text{UGOODTF}_n{*}\text{UTF2}_n \quad \text{(Model 9)}$$

Here, UGOODTF_n in models (8) and (9) is the term frequency (proportion) of the unique "good words" that appear in the body of the nth web page. Some other possible models were also considered but did not perform as well as the model in Eq. (1). Due to space limitations, we did not include them here.

We use maximum likelihood estimation to estimate these models and compare them to the proposed model. The specification of all models and estimation results are also showed in Table 5. Likelihood Ratio Tests and AIC are used to compare nested models and non-nested models, respectively

Table 5
Model assessment

	Model 1 (Intercept+ HEADBAD)	Model 2 (Intercept+ HEADBAD+ HEADGOOD)	Model 3 (Intercept+ HEADBAD+ HEADGOOD+ HEADBAD* HEADGOOD)	Model 4 (Intercept+ HEADBAD+ UTF4+UTF3 + UTF2)	Model 5 (Model 4+ HEADGOOD+ GOODTF)	Model 6 Model 5+ GOODTF* (UTF4+UTF3+ UTF2))	Model 7 (Model 5+ GOODTF*UTF4+ GOODTF*UTF3+ GOODTF*UTF2)	Model 8 (Model 4+ HEADGOOD+ UGOODTF)	Model 9 (Model 8+ UGOODTF*UTF4+ UGOODTF*UTF3+ UGOODTF*UTF2)
Intercept	−0.421**	−0.312*	−0.312**	−1.555**	−1.157**	−1.128**	−1.134**	−1.075**	−1.1952**
	(0.056)	(0.083)	(0.083)	(0.105)	(0.117)	(0.129)	(0.134)	(0.171)	(0.144)
HEADBAD	0.787**	0.894**	0.877**	0.355**	0.373**	0.374**	0.370**	0.266	0.325*
	(0.083)	(0.136)	(0.139)	(0.111)	(0.142)	(0.141)	(0.143)	(0.189)	(0.135)
HEADGOOD		−1.329**	−1.468*		−0.573	−0.568	−0.577	−0.822	−0.810
		(0.319)	(0.460)		(0.643)	(0.647)	(0.650)	(0.778)	(0.535)
HEADBAD* HEADGOOD			0.192 (0.438)						
GOODTF					−112.156**	−120.160**	−118.023**		
					(14.686)	(21.710)	(26.676)		
UGOODTF								−190.073**	−142.203**
								(42.029)	(43.021)
UTF4				327.021**	329.667**	305.991**	301.487**	304.286*	460.516**
				(60.511)	(76.895)	(90.692)	(98.218)	(124.365)	(150.037)
UTF3				120.561**	232.136**	232.008**	232.951**	198.250**	240.958**
				(17.701)	(30.378)	(30.583)	(33.028)	(36.949)	(34.604)
UTF2				148.290**	145.886**	144.892**	146.562**	160.660**	146.847**
				(20.13)	(22.768)	(22.886)	(25.132)	(31.672)	(25.981)
GOODTF*(UTF4 + UTF3 +UTF2)						185.038 (434.519)			
GOODTF*UTF4							456.775		
							(3579.568)		
GOODTF*UTF3							100.954		
							(3109.747)		
GOODTF*UTF2							−341.110		
							(3545.214)		
UGOODTF*UTF4									−73868.749
									(37941.51)
UGOODTF*UTF3									−12525.315*
									(5767.923)
UGOODTF*UTF2									6223.837
									(5366.132)
Log−likelihood	−403.1398	−371.661578	−371.46587	−197.3373	−128.992761	−128.8474756	−128.8322595	−140.229996	−135.63533
Training									
Sensitivity	54	54	54	83	90	90	91	90	91
Specificity	90	95	95	93	93	96	96	95	96

* Denotes significance at 0.05 level.

** Denotes significance at 0.01 level.

[1,14]. The sensitivity and specificity are also compared as secondary measures across models. Table 5 shows all parameters used in the above nine models. For example, if we look at the third column of Table 5, it shows that parameter values used for Model (2). That is, δ_0=−0.315945, δ_1=0.893671, δ_2=−1.329621. The log-likelihood is −371.661578. If we use model (2) to learn a concept from the training data set, the sensitivity will be 54 and the specificity will be 95.

From Table 5 we can see that Models (1)–(3), and Models (4), (5), and (7) are nested models, respectively. Models (4), (8) and (9) are also nested models. We use Likelihood Ratio Tests – a widely used approach for comparison of different nested models [14, p. 303–306]–to compare these nested models. The test statistic is equal to $-2(LL_R–LL_U)$, where LL_R is the log-likelihood of restricted model and LL_U is the log-likelihood of unrestricted model. This test statistic is approximately Chi-square distributed with the degree of freedom equal to the number of restrictions. If we compare Models (1)–(3) by using Likelihood Ratio Tests, it is easy to see that Model (2) has the highest goodness of fit. Compare the proposed model (which is Model (7)) to Models (4) and (5), the test statistics are 137.01 (>14.07 with 7 degrees of freedom at significance level 0.05) and 0.321 (<7.81 with 3 degrees of freedom at significance level 0.05), respectively. Therefore, the proposed model has better goodness of fit than Model (4). Although the restrictions fail to be rejected when we compare Model (5) to Model (7), Model (7) has high sensitivity and specificity. Hence Model (7) is chosen against Model (5). Similarly, Model (8) has better goodness of fit than Model (4) (the test statistic is 114.21>5.99 with 2 degrees of freedom at significance level 0.05) and Model (9) is preferred against Model (8) (the test statistic is 9.1893>7.81 with 3 degrees of freedom at significance level 0.05). Thus Model (9) is chosen among these three models. Since the proposed model (Model (7)), Models (2), (6), and (9) are non-nested models, Akaike Information Criterion (AIC) is used to compare these four models [1]. AIC is equal to $-2(LL-P)/N$, where LL is the log-likelihood, P is the number of parameters and N is the number of observations. The model with lower AIC has higher goodness of fit. The AIC of the proposed model is 0.36755 and the AIC of the other three models are 0.9991, 0.36226, and 0.3857, respectively. Hence the proposed model has a lower AIC than Models (2) and (9). Although the AIC of Model (6) is slightly lower than that of the proposed model, the proposed model has higher sensitivity and specificity. Therefore, our proposed model is preferred. We also use Test sample 2: chat room name data to test the predictive ability of our proposed model. The sensitivity and specificity are 100 (0 error out of 1 "bad" sites) and 100 (0 error out of 221 "good" sites), respectively. Hence, our proposed model also has good predictive ability.

Appendix B. Determination of the size of the training data

We used an analytical approach to determine the size of the training data in this study. In the statistics literature [10,15,21], the widely used formulae that frequentists use to determine the sample size for the normal distribution is given by

$$N \geq \frac{4*\sigma^2*Z^2_{1-\alpha/2}}{l^2},$$

where N is the sample size, σ^2 is the variance, and $Z_{1-\alpha/2}$ is the $(1-\alpha/2)$ - percentile of the standard normal distribution. The sample size N satisfying the above condition guarantees that a $100*(1-\alpha)\%$ confidence interval will be of total length l. So given our standard normal distribution assumption of the error term in our Probit model, standard 95% confidence interval level (α=5%) with short length 0.2, the minimum sample size that satisfies the above condition for our model is 384. Since the sample size in our training sample is 750 which is far more than 384, we are confident that we have sufficient data for model estimation purposes.

References

[1] H. Akaike, Information theory and an extension of the maximum likelihood principle, in: B. Petrov, F. Csake (Eds.), Second international symposium on information theory, Akadémiai Kiado, Budapest, 1973.

[2] M. Balabanovic, Y. Shoham, Fab: content-based, collaborative recommendation, Communications of the ACM 97 (3) (1997) 66–72.
[3] D. Boley, M. Gini, R. Gross, E.-H. Han, K. Hastings, G. Karypis, V. Kumar, B. Mobasher, J. Moore, Partitioning-based clustering for web document categorization, Decision Support Systems 27 (1999) 329–341.
[4] G. Boone, Concept features in Re: agent, an intelligent email agent, Proceedings of the second International Conference on Autonomous Agents, Minneapolis, 1998.
[5] J.L. Center Jr., Bayesian classification using an entropy prior on mixture models, Proceedings 19th International Workshop on Bayesian Inference and Maximum Entropy Methods in Science and Engineering, 1999, pp. 42–70.
[6] P. Cheeseman, J. Stutz, Bayesian classification (AutoClass): theory and results, in: U.M. Fayyad, G. PiatetskyShapiro, P. Smyth, R. Uthurusamy (Eds.), Advances in knowledge discovery and data mining, AAAI Press, Menlo Park, CA, 1996, pp. 158–180.
[7] L.F. Cranor, P. Resnick, D. Gallo, Technology inventory: a catalog of tools that support parents' ability to choose online content appropriate for their children. Prepared for the Internet Online Summit: Focus on Children, December 1997, revised September, 1998 for America Links Up. http://www.research.att.com/projects/tech4kids/.
[8] M. Craven, D. Freitag, A. McCallum, T. Mitchell, K. Nigam, C.Y. Quek, Learning to extract symbolic knowledge from the world wide web. Technical Report, Carnegie Mellon University, 1997.
[9] C. De Loupy, P. Bellot, M. El-Beze, P.F. Marteau, Query expansion and classification of retrieved documents, Seventh text retrieval conference (TREC-7), 1998, pp. 443–450.
[10] M.M. Desu, D. Raghavarao, Sample size methodology, Academic Press, Boston, 1990.
[11] W. Ding, Implementing parental control on access to inappropriate web pages: Filtering by a Good word/Bad word Probit analysis, Master of Philosophy thesis, Carnagie Mellon University, 2000.
[12] S.T. Dumais, Latent semantic indexing and TREC-2, in: D. Harman (Ed.), The second text retrieval conference (TREC2), 1994, pp. 105–116.
[13] A. Gibbons, Algorithmic graph theory, Cambridge University Press, 1985.
[14] W.H. Greene, Econometric analysis, Third edition, Prentice-Hall, 1997, pp. 882–885.
[15] T. Hagerup, C. Rub, A guided tour of chernoff bounds, Information Processing Letters 33 (1989) 305–308.
[16] J.A. Hartigan, M.A. Wong, A K-means clustering algorithm: algorithm AS 136, Applied Statistics 28 (1979) 126–130.
[17] J. Herlocker, J. Konstan, J. Riedl, Explaining collaborative filtering recommendations, Proceedings of ACM 2000 Conference on Computer Supported Cooperative Work, 2000, pp. 241–250.
[18] U. Heuser, W. Rosenstiel, Automatic generation of local Internet catalogues using hierarchical radius-based competitive learning, Proceedings 14th European Conference on Artificial Intelligence, 2000, pp. 306–310.
[19] V. Jacob, R. Krishnan, Y.U. Ryu, R. Chandrasekaran, S. Hong, Filtering objectionable internet content, Proceedings of the 20th International Conference on Information Systems, 1999.
[20] K. Jain, R.C. Dubes, Algorithms for clustering data, Prentice Hall, Englewood Cliffs, NJ, 1988.
[21] L. Joseph, P. Belisle, Bayesian sample size determination for normal means and differences between normal means, The Statistician 46 (2) (1997) 209–226.
[22] T.K. Landauer, P.W. Foltz, D. Laham, An introduction to latent semantic analysis, Discourse Processes 25 (1998) 259–284.
[23] Y. Lashkari, M. Metral, P. Maes, Collaborative interface agents, Proceedings of the Twelfth National Conference on Artificial Intelligence, Seattle, AAAI Press, 1994, pp. 444–450.
[24] J. Lasica, Ratings today, censorship tomorrow, Salon Magazine, 1997 (July).
[25] P.Y. Lee, S.C. Hui, A.C.M. Fong, Neural network for web content filtering, IEEE Intelligent Systems 17 (5) (2002) 48–57.
[26] H. Lieberman, Letizia: an agent that assists web browsing, Proceedings of the International Joint Conference on Artificial Intelligence, Montreal, 1995.
[27] P. Maes, R. Kozierok, Learning interface agent, Proceedings of Eleventh National Conference on Artificial Intelligence, MIT Press, 1993.
[28] M.B. Menhaj, F. Delgosha, A soft probabilistic neural network for implementation of Bayesian classifiers, Proceedings International Joint Conference on Neural Networks, 2001, pp. 454–458.
[29] D. Midine, D.J. Spiegelhalter, C.C. Taylor, Machine learning, neural and statistical classification, Ellis Horwood, New York, 1994, edited collection.
[30] J. Miller, P. Resnick, D. Singer, Rating service and rating systems (and their machine readable descriptions), http://w3.org/PICS/services.html, May, World Wide Web Consortium, 1996.
[31] R.A. Mollineda, E. Vidal, A relative approach to hierarchical clustering, pattern recognition and applications (frontiers in artificial intelligence and applications Vol.56) p. viii+287, 19–28, 2000.
[32] M. Morita, Y. Shinoda, Information filtering based on user behavior analysis and best match text retrieval, Proceedings of the 17th ACM Annual International Conference on Research and Development in Information Retrieval (SIGIR'94), Springer-Verlag, Dublin, Ireland, 1994, pp. 272–281.
[33] M.J. Pazzani, J. Muramatsu, D. Billus, Syskill & Webert: identifying interesting web sites, Proceedings of Thirteenth National Conference on Artificial Intelligence, Portland, AAAI Press, 1996, pp. 54–61.
[34] J.R. Quinlan, C4.5: programs for machine learning, Morgan Kaufmann Publishers, San Mateo, CA, 1993.
[35] A.M. Rashid, I. Albert, D. Cosley, S.K. Lam, S. McNee, J.A. Konstan, J. Riedl, Getting to know you: learning new user preferences in recommender systems, Proceedings of the 2002 International Conference on Intelligent User Interfaces, San Francisco, CA, 2002, pp. 127–134.
[36] P. Resnick, Filtering information on the Internet, Scientific American, 1997 (March).

[37] P. Resnick, J. Miller, PICS: internet access controls without censorship, Communications of the ACM (1996 (October)) 87–93.
[38] G. Salton, M.J. McGill, Introduction to modern information retrieval, McGraw-Hill, New York, 1993.
[39] S. Sarkar, R. Sriram, Bayesian models for early warning of bank failures, Management Science 47 (11) (2001) 1457–1475.
[40] B. Shapira, P. Shoval, U. Hanani, Experimentation with an information filtering system that combines cognitive and sociological filtering integrated with user stereotypes, Decision Support Systems 27 (1999) 5–24.
[41] B. Sheth, P. Maes, Evolving agents for personalized information filtering, Proceedings of the Ninth IEEE Conference on Artificial Intelligence for Applications, 1993.
[42] B. Sierra, N. Serrano, P. Larranaga, E.J. Plasencia, I. Inza, J.J. Jimenez, P. Revuelta, M.L. Mora, Using Bayesian networks in the construction of a bi-level multi-classifier: a case study using intensive care unit patients' data, Artificial Intelligence in Medicine 22 (3) (2001) 233–248.
[43] Survey, National Public Radio, the Kaiser Family Foundation, and Harvard's Kennedy School of Government, http://www.npr.org/programs/specials/poll/technology, 2000.
[44] M.A. Tanner, Tools for statistical inference, Springer-Verlag, 1996.
[45] Z. Wang, J. Li, G. Wiederhold, O. Firschein, Classifying objectionable websites based on image content, IDMS (1998) 113–124.
[46] J. Weinberg, Rating the net. http://www.msen.com/~weinberg/rating.htm, 1997.
[47] M.R. Wulfekuhler, W.F. Punch, Finding salient features for personal web page categories, Proc. of 6th International World Wide Web Conference, 1997 (April), http://www.scope.gmd.de/info/www6/technical/paper118/paper118.html.

Jonathan P. Caulkins is Professor of Operations Research and Public Policy. He specializes in mathematical modeling of social policy problems, particularly those pertaining to drugs, crime, violence, deviance, public health, and prevention issues, but also works on software quality, rating and evaluation problems, and optimal dynamic control applications. Caulkins earned a BS and MS in Systems Science and Mathematics from Washington University, an MS in Electrical Engineering and Computer Science, and a PhD in Operations Research from MIT.

Wenxuan Ding is an assistant professor in Department of Information and Decision Sciences and of Computer Science at University of Illinois, Chicago. She received a BS in Computer Science, an MS in Computer Science from National University of Singapore, a Master of Philosophy in Public Policy and Management, a PhD in Information Technology and Cognitive Science from Carnegie Mellon University.

George T. Duncan is Professor of Statistics in the Heinz School of Public Policy and Management at Carnegie Mellon University, where his research centers on information technology and social accountability. He chaired the Panel on Confidentiality and Data Access of the National Academy of Sciences, resulting in the book, *Private lives and public policies: confidentiality and accessibility of government statistics*. He is a Fellow of the American Statistical Association, an elected member of the International Statistical Institute, and a Fellow of the American Association for the Advancement of Science.

Ramayya Krishnan is the W.W. Cooper and Ruth F. Cooper Professor of Information Systems at Carnegie Mellon University. He has a B. Tech in Mechanical Engineering from the Indian Institute of Technology, Madras, an MS in Industrial Engineering and Operations Research, and a PhD in Management Science and Information Systems from the University of Texas at Austin. He is an International Research Fellow of the International Center for Electronic Commerce in Korea and a Visiting Scientist at the Institute for Information Systems at Humboldt University (Germany). He is faculty chair of the university's Masters of Information Systems Management program. Krishnan's current research interests lie in problems that arise at the interface of technology, business and policy aspects of internet-enabled systems.

Eric Nyberg is an Associate Professor at the Heinz School and the Language Technology Institute of the School of Computer Science.