INFERRING WEB PAGE RELEVANCE FROM HUMAN-COMPUTER INTERACTION LOGGING

Vincenzo Deufemia, Massimiliano Giordano, Giuseppe Polese and Genoveffa Tortora
Università di Salerno, Via Ponte don Melillo, Fisciano, SA, Italy

Keywords: Learning from User Behavior, Implicit Feedback, Mouse Tracking, Search Engines, Display Time, Logging Information.

Abstract: Quality of search engine results often do not meet user's expectations. In this paper we propose to implicitly infer visitors feedbacks from the actions they perform while reading a web document. In particular, we propose a new model to interpret mouse cursor actions, such as scrolling, movement, text selection, while reading web documents, aiming to infer a *relevance* value indicating how the user found the document useful for his/her search purposes. We have implemented the proposed model through light-weight components, which can be easily installed within major web browsers as a plug-in. The components capture mouse cursor actions without spoiling user browsing activities, which enabled us to easily collect experimental data to validate the proposed model. The experimental results demonstrate that the proposed model is able to predict user feedbacks with an acceptable level of accuracy.

1 INTRODUCTION

The goal of Information retrieval (IR) is to find relevant documents as response to users queries. In finite and controlled information sources, such as document collections, the vector space model based on the popular Term Frequency - Inverse Document Frequency ($Tf - Idf$) measure is traditionally used to find relevant documents (Salton and Buckley, 1988). However, this approach is impractical in huge and uncontrolled environments like the Web, where large quantities of resources constantly compete to draw user attention. The sheer size of the Web does not allow for presenting the entire set of related documents to the user. In such an environment the quality of documents plays an important role, and measuring it merely based on page contents is difficult and may also be subjective.

One way to derive an objective measure of documents' usefulness is to exploit the linked structure of the Web. PageRank is the most famous method using link structure analysis (Page et al., 1999). The idea behind PageRank algorithm is to exploit the macro-scale link structure among pages in order to capture the popularity of documents, which can indirectly be interpreted as an index of their quality. According to this approach, the popularity of a page is determined on the basis of the size of a hypothetical user stream coming to the page. However, link-based algorithms have currently many disadvantages (Mandl, 2006). For example, they are vulnerable to spamming, and links may have several meanings or purposes.

With the advent of Web 2.0, *social bookmarking* systems have started showing the potential for improving search capabilities of current search engines. In these systems, the popularity of a Web page is calculated as the total number of times it has been bookmarked, which is interpreted as the number of users voting for the page.

There are several differences between "classic" ranking systems like PageRank, and explicit ones based on explicit feedbacks (Yanbe et al., 2007). Explicit ranking systems capture the popularity of resources among content consumers (page readers), while PageRank is a result of author-to-author evaluation of Web resources. Generally, explicit rank is more dynamic than PageRank, and *social bookmarking* systems often ensure shorter time for pages to reach their popularity peaks (Golder and Huberman, 2006). However, despite many advantages of *social bookmarking* services, relying on them alone is currently still not possible due to the insufficient amount of bookmarked pages available for arbitrary queries. Furthermore, explicit ranking is subjective, since users need to explicit vote a web content to rate it, and not all the web users are keen on voting

Deufemia V., Giordano M., Polese G. and Tortora G..
INFERRING WEB PAGE RELEVANCE FROM HUMAN-COMPUTER INTERACTION LOGGING.
DOI: 10.5220/0003938406530662
In *Proceedings of the 8th International Conference on Web Information Systems and Technologies* (WEBIST-2012), pages 653-662
ISBN: 978-989-8565-08-2

each site they visit. Thus, despite the rapid growth in the number of bookmarked pages, the combination of link structure-based and *social bookmarking*-based page ranking measures seems to be currently an optimal strategy.

Alternatively, methods that are able to implicitly capture user interests are potentially more useful, since there is no noise in the ranking process introduced by subjective evaluations (Agichtein et al., 2006; Fox et al., 2005). Thus, we have started exploiting methods for logging user interaction actions in order to derive an implicit index expressing the web page usefulness with respect to user interests. In particular, we propose a new model to interpret mouse cursor actions, such as scrolling, movement, text selection, while reading web documents, aiming to infer a *relevance* value indicating how the user found the document useful for his/her search purposes (Chen et al., 2001; Mueller and Lockerd, 2001).

We have embedded the proposed model in a ranking system for the web. In particular, we have implemented the YAR (Yet Another Ranker) system, which re-ranks the web pages retrieved by a search engine based on the relevance values computed from the interaction actions of previous visitors. YAR ha been implemented by means of light-weight components, which can be easily installed within major web browsers as a plug-in (we used it experimentally with Google, but any other search engine could be easily adapted). The implemented components capture mouse cursor actions without spoiling user browsing activities, which enabled us to easily collect experimental data to validate the proposed model. The experimental results demonstrate that the proposed model is able to predict user feedbacks with an acceptable level of accuracy.

The paper is organized as follows. Section 2 describes the metrics for deriving the web page relevance from mouse tracking logging data. An implementation of the proposed metrics in the context of ranking systems is presented in Section 3. Section 4 presents an experimental evaluation with analysis of the results. A comparison with related work is described in Section 5. Finally, conclusions and future work are discussed in Section 6.

2 THE METRICS FOR WEB PAGE RELEVANCE

In order to compute the web page relevance value we consider several metrics. The application of all these metrics will be used to produce a value between 1 and 5, as usually done in *social bookmarking* systems. In particular, we have defined the following metrics:

- permanence time,
- reading rate,
- scrolling rate.

The overall rate is obtained through a weighted sum of the considered metrics. Linear regression has been used to find the weights for metrics that best explain the observed user feedback.

2.1 Permanence Time

The Permanence Time (PT) is defined as the difference between the loading and the unloading time of a web page. Obviously, PT is heavily influenced by the way the user reads a text within a document and by the number of words composing it. Several studies prove that there are different ways of reading a text, each corresponding to a different speed, also depending on reader's language and age (Hunziker, 2006). Rates of reading are measured in words per minute (wpm), and include *reading for memorization* (less than 100 wpm), *reading for learning* (100 – 200 wpm), *reading for comprehension* (200 – 400 wpm), and *skimming* (400 – 700 wpm).

In general, being aware of the reading style seems to be essential in order to correctly relate the time the user spends on a page with his/her hypothetical interest. In spite of all those kind of different reading strategies, the way user reads on the web seems to be different with respect to the way they read a printed text. Usually, web users rapidly find key elements of a document, and they usually highlight sections, paragraphs, and keywords by using the mouse cursor. The web user only reads a small portion of a web page, usually between the 20% and 28% of it (Nielsen, 2008).

Furthermore, experimental data show that web pages containing from 30 to 1250 words are read shallowly, and that the estimated time a user will stay on the web page, before making a decision about its usefulness, is at least 25 seconds plus 4.4 seconds for each block of 100 words (Nielsen, 2008).

Starting from these results, we define PT as:

$$PT = \begin{cases} pT_m \cdot (3/Tref_m) + 1 & \text{if } pT_m \leq Tref_m \\ pT_m \cdot (2/Tref_{max}) + 4 & \text{if } pT_m > Tref_m \end{cases} \tag{1}$$

where:

- pT_m is the *average permanence time* and it is defined as $pT_m = (pw \cdot 0.044) + 25$, with pw representing the number of words composing the page;
- $Tref_{max}$ is the *maximum reference time* and it is

defined as $Tref_{max} = (pw/150)60$. We assume the *reading for learning* rate (about 150 wpm in the average case) as its lower bound;

- $Tref_m$ is the *average reference time*.

$Tref_m$ is defined to support fast reading strategies, typical of web users, and when the number of words is between 30 and 1250 (Nielsen, 2008). Thus, for this kind of pages the $Tref_m$ is defined as the *average permanence time*. However, for longer documents this equality might be inaccurate. In these cases we redefine $Tref_m$ as:

$$Tref_m = pw/v_{lett} \cdot 60 \qquad (2)$$

where v_{lett} is the reading speed rate corresponding to the selected reading strategy. As default, we assume an average rate of $v_{lett} = 300wpm$, which corresponds to the *reading for comprehension* strategy.

In conclusion, we associate an average relevance value of 3 to a document on which the user spends a time equal to $Tref_m$. For a shorter permanence time the relevance value is computed by using the average time as upper bound. In case of permanence time greater than the average time, we use $Tref_{max}$ as upper bound. In this way, the metric is more sensible to the different reading strategies.

Notice that the resulting value for *PT* is in the range $[1,\infty]$. However if it is greater than 5, we reduce it to 5, because we have empirically verified that above 5 the user interest does not increase considerably.

2.2 Reading Rate

A common user activity on the web is text filtering. As shown in many usability studies performed by using *eye tracking*, users are often interested in some portion of the text, and only in some of its contents (Nielsen, 2006). S/he follows a "standard" reading schema, called the "F" reading pattern. Thus, by analyzing the mouse activities, we can review the same reading pattern, and use it to understand how much the page is useful to the user.

In particular, experimental data show that many users navigate through the page by pointing with the mouse cursor near the rows they find interesting (Huang et al., 2011). However, this behavior is not common to all users. Alternatively, some users might highlight text either to facilitate reading, to copy it, or just to print the selected portion. Obviously, these can all be interpreted as measures of interest. Thus, we can use such mouse actions to derive a measure, called *Reading Rate* (RR), estimating the amount of text the user reads in the document, which is computed as:

$$RR = 5 \cdot \left(\frac{rw + sw}{pw}\right) + 1 \qquad (3)$$

where:

- *rw* is the number of words followed by the mouse cursor;
- *pw* is the total number of words in the document;
- *sw* is the total number of selected words.

Notice that the number of words followed by the cursor is added to the number of words selected during the page exploration. In fact, often the user moves the mouse over each selected word to facilitate reading. This can also be taken as a further demonstration of user interest.

The resulting value for *RR* is also in the range $[1,\infty]$. Thus, we normalize it in the range $[1,5]$ for reasons similar to those used for *PT* value.

2.3 Scrolling Rate

During a navigation session the web user might not necessarily read all the contents inside a page. Often, users scroll the page looking for interesting contents, or merely to have a complete overview of it.

The main reasons to scroll a page are:

- to navigate from the current section to the next one;
- to find a paragraph, or just some more interesting keywords;
- to skip the entire content of the page and reach a link to the next one, like for the classic End User Licensing Agreements (EULA) pages.

We believe that scrolling should be considered as a measure of interest for page contents. In other words, a relevant scrolling activity might witnesses that the user is interacting with the web page, and that s/he has not left the browser idle, because s/he is doing some other activity, leaving the value of the permanence time grow inappropriately. On the other hand, a highly frequent scrolling activity might convey a low interest, because the user might be skimming over the page without finding contents of interest to him/her.

In these cases, we apply a penalty to the relevance value. To this end, we need to measure the scrolling activities during the navigation. We call this measure Scrolling Rate (SR) and define it as the following normal distribution:

$$SR = 4 \cdot \left(exp\left[-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2\right]\right) + 1 \qquad (4)$$

where

- $x = (N_{scroll}/PT) \cdot 60$ is the scrolling frequency expressed in terms of number of scrolls (N_{scroll}) per minute;
- μ is the mean value (the peak of the curve), which represents the scrolling frequency in case of high interest for page contents. We have empirically determined an optimal value of 25 for this parameter;
- σ^2 is the variance and it represents the range of scrolling frequencies revealing some interest for page contents. We have empirically determined an optimal value of 7.

Thus, the function SR contributes to increase the relevance value when the scrolling frequency is in the range 25 ± 7, as also shown in Figure 1. Beyond such range there is little interest, either because of too fast or reduced scrolling.

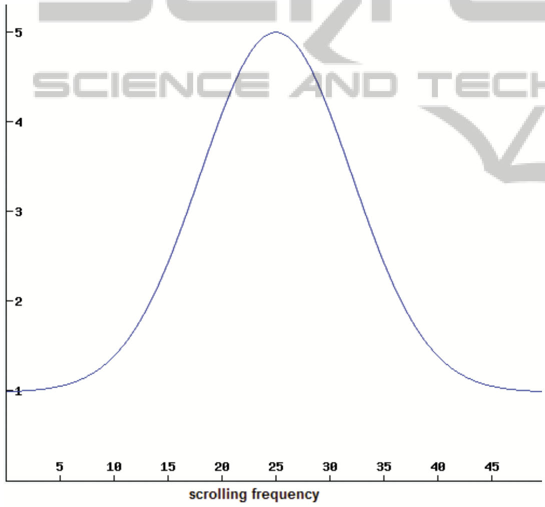


Figure 1: The SR function.

3 AN IMPLEMENTATION OF THE PROPOSED METRICS

Implicit feedbacks have a wide range of applications. In this section, we present an implementation of the proposed metrics in the context of web page ranking. In particular, we present the YAR system whose architecture is depicted in Figure 2. It is based on a client/server model, where data concerning user interactions are collected on the client side by the *Logger*, and evaluated on the server side through the *Log Analyzer*. The *Logger* is responsible for "being aware" of the user's behavior while s/he browses web pages, and for sending information related to the captured events to the server-side module. The latter is responsible for analyzing the collected data, applying the metrics, and deriving relevance values to be successively used for ranking purposes.

The following subsections provide details on the modules composing the YAR system.

3.1 The Logging Module

One way to collect data concerning users interactions is to track their eyes' movements. However, this would require the use of expensive tools, which would make it difficult to run large-scale simultaneous experiments. Nevertheless, it has been shown that similar results can also be inferred by tracking mouse movements. In fact, it has been experimentally proved that in more than 75% of cases the mouse cursor closely approximates the eye gaze (Chen et al., 2001; Mueller and Lockerd, 2001). This important result suggests that *mouse tracking* might replace eye tracking, allowing the extraction of many useful information about the user interest regarding a web page. This finding is also confirmed by a recent study on the correlation between cursor and gaze position on search result pages (Huang et al., 2011).

In light of the above arguments, our logging module tracks user interaction actions through several devices, but it does not perform eye tracking. In particular, the logging module tracks the overall and the effective permanence time over a web page, mouse cursor movements, page scrolling events, text selection, and so forth. It is based on the *AJAX* technology (Murray, 2006) to capture and log user's interactions with a web system through a pluggable mechanism, which can be installed on any web browser. Thus, it does not require modifications to the web sites, or any other legacy browser extensions. In particular, the architecture of the *Logger* is graphically represented in Figure 3.

It is structured in the following three main subcomponents:

- Page handler: it handles page loading and unloading events.
- Mouse handler: it handles mouse events.
- Text handler: it handles keyboard related events.

These generic handlers could be overridden with ad-hoc specializations letting the system filter different kinds of events, so that it can be adapted to many different application domains.

An important property of the *Logger* component is flexibility. The *JavaScript* code for event capturing may be dynamically configured in order to record several kind of events occurring during the user navigation. Each class of events is handled by a specific

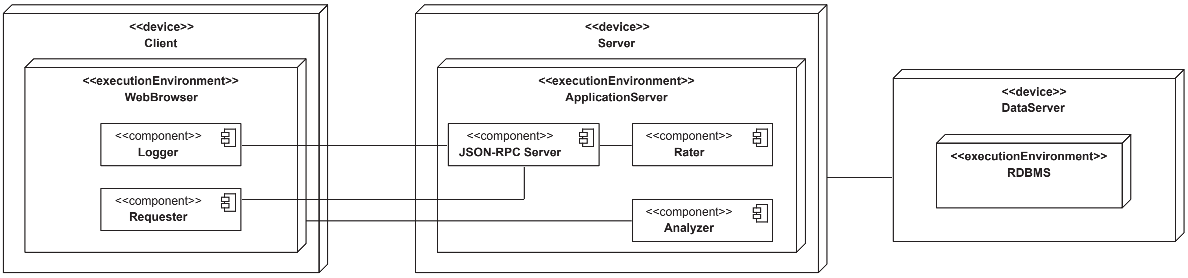


Figure 2: The YAR system architecture.

handler. Among the parameters that can be configured for the logger we have:

- list of events to capture;
- sub-set of attributes for each event;
- sections of the web pages (*divs* or table cells) to be monitored as event sources;
- time interval between two data transmissions from the client to the server;
- sensitivity for mouse movements (short movements are not captured).

By acting on these parameters we have the possibility to affect the size of the collected data.

3.2 The Log Analyzer

The *Log Analyzer* is a server-side module providing two main functionalities: *rating* and *reporting*. The former is accomplished by the *Rater*, which rates the currently opened documents by using data that the *Logger* has collected on the server during navigational sessions. To this end, the metrics adopted for ranking depend on the application domain. For example, we can derive metrics for web search applications, metrics to evaluate usability of software systems, or to evaluate the satisfaction of a user while using an automatic *Help Desk* system or an *E-testing* system. The overriding mechanism used to specialize the *Rater* is illustrated in Figure 4.

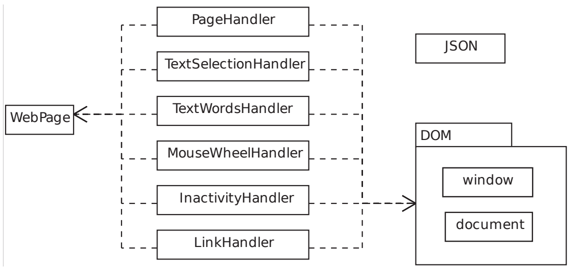


Figure 3: The logging module.

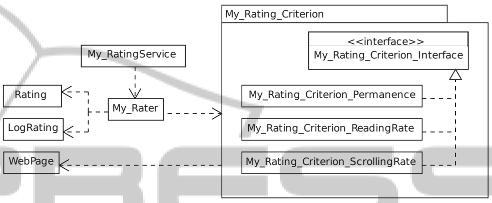

My_Rating_Criterion
<<interface>>
My_Rating_Criterion_Interface
My_RatingService
Rating
My_Rater
LogRating
WebPage
My_Rating_Criterion_Permanence
My_Rating_Criterion_ReadingRate
My_Rating_Criterion_ScrollingRate

Figure 4: The rater's overriding mechanism.

The *reporting* subsystem ensures the access to the gathered data by means of domain specific visual metaphors. In the web search context, this module uses a simple graphical pattern to show the rank produced by the ranker components, and mixes such results with those provided by the underlying search engine. All the reporting facilities are accessible through a web-based application or as a service, as in the case of information ranking.

3.3 Integration into SERPs

Thanks to the availability of reporting services, we can ask the system to provide the relevance value for each link already visited by other users. Thus, apart from collecting human computer interaction data, and calculating/updating the implicit rank, we integrate the rank information within a Search Engine Report Page (*SERP*). In particular, we show this in the context of *Google* search engine, but any other search engine could be used.

The integration with *SERP* is done by means of the same technology used to log user interaction data. We prepared a JavaScript function directly installed on the user browser in the same way as we integrated the logging facilities. However, in this case, instead of logging user interactions, the script scans the *SERP* and inquires the implicit rank for each link it contains. Finally, the script modifies the Document Object Model (*DOM*) of the web page in order to show the rank beside each result. In particular, the rank is shown by using a simple visual metaphor by depicting as many Star" symbols as the rank value. Thus, the system associates from 0 to 5 star symbols to each link. After each search, YAR shows the new page instead of the standard *SERP* page, and all the links originally found by the search engine are re-ranked by using the ranking information, if available.

In Figure 5 we can see the *Google SERP* page (Figure 5(a)) and the YAR re-ranked page (Figure 5(b)). In particular, the two pages result from the query: "html5 xhtml2". Notice that pages that were in the middle of the original *SERP*, after YAR re-ranking are positioned on the top of the list. This means that they better captured the interest of users who had previously visited them.

4 EXPERIMENTAL EVALUATION

In this section we validate the proposed implicit feedback measure through a user study. The latter has a twofold goal. On one hand, we need to understand how the single metrics should be weighted when deriving the global implicit feedback measure. On the other hand, we need to evaluate the effectiveness of such measure. One way to do this, is to compare the computed implicit feedbacks with respect to the user provided ones.

4.1 Evaluation Method

The evaluation was accomplished by two independent groups of participants. They were prescribed the same tasks. However, the first group (20 participants) exercised the system in a basic configuration, and the results where used to derive an optimal parameter settings for the proposed implicit feedback method. Based on such settings, the second group (6 participants) produced results that were used to validate the effectiveness of the proposed measure.

4.1.1 Participants

We selected twenty-six participants between 22 and 31 years old. Sixteen of them had a bachelor in computer science, four a technical high school degree, two a gymnasium high school degree, two a master in linguistics, and two a bachelor in chemistry. Nine of them were female and seventeen were male. All the participants had sufficient computer and World Wide Web experience, and an average of 7.6 years of searching experience. Each of them underwent a week time period of search experiments.

4.1.2 User Tasks

In order to evaluate our system, we asked users to perform web searches and to explicitly rate the usefulness of the retrieved web documents. Then, we needed to compare such rates with those implicitly derived through the proposed approach. However, given the magnitude of the web, to have a significant amount of experimental data, we needed to narrow the scope of user searches in order to guide them towards a restricted set of web contents. This has been accomplished by assigning users specific webquests (Dodge, 1995). A webquest is a short description of a specific topic, on which a user should write an essay by mainly investigating through web sources. They are frequently used in e-learning contexts to give learners a clear purpose and objective when searching through web sources of knowledge.

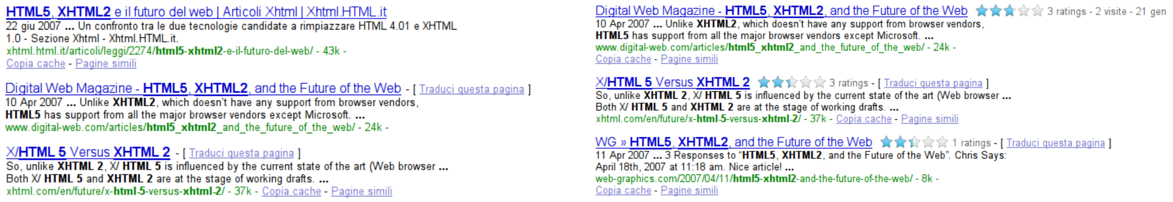
Ten webquests in italian language were prepared for the experiment. They regarded well-known topics such as for example retirement plans, anxiety, and coffee.

Each participant was requested to select three out of the ten available webquests, and to solve them. For each visited page, they were requested to express a vote representing how they judged the page useful to solve the specific webquest. The votes were expressed in a Likert scale [1, 5] where 1 represented *not useful* and 5 *useful*.

4.1.3 Instruments and Procedure

Each participant had his/her own computer on which we installed the YAR software. The latter also included a module for expressing a vote when leaving a visited web page.

Other than the twenty-six participants, six computer science students, one undergraduate and five graduate, participated in the organization and supervision of experiments. The undergraduate student prepared all the webquests as part of her bachelor project, whereas each graduate student was requested to select four participants, and had the responsibility to conduct experiments with them in order to derive an optimal system tuning. After one week of experiments, they had a meeting with us to analyze and discuss experimental data, and to reach an agreement on the proper parameter settings to be used for the experiments with the remaining six subjects. The latter were selected among computer science students attending a graduate course on web engineering. They also worked one week, after which they had a final meeting with us to summarize and analyze the experimental results.



HTML5, XHTML2 e il futuro del web | Articoli Xhtml | Xhtml.HTML.it
22 giu 2007 ... Un confronto tra le due tecnologie candidate a rimpiazzare HTML 4.01 e XHTML 1.0 - Sezione Xhtml - Xhtml.HTML.it.
xhtml.html.it/articoli/leggi/2274/html5-xhtml2-e-il-futuro-del-web/ - 43k -
Copia cache - Pagine simili

Digital Web Magazine - HTML5, XHTML2, and the Future of the Web - [Traduci questa pagina]
10 Apr 2007 ... Unlike XHTML2, which doesn't have any support from browser vendors, HTML5 has support from all the major browser vendors except Microsoft. ...
www.digital-web.com/articles/html5_xhtml2_and_the_future_of_the_web/ - 24k -

X/HTML 5 Versus XHTML 2 - [Traduci questa pagina]
So, unlike XHTML 2, X/ HTML 5 is influenced by the current state of the art (Web browser ...
Both X/ HTML 5 and XHTML 2 are at the stage of working drafts. ...
xhtml.com/en/future/x-html-5-versus-xhtml-2/ - 37k - Copia cache - Pagine simili

(a) Google rank.

Digital Web Magazine - HTML5, XHTML2, and the Future of the Web ★★★☆☆ 3 ratings - 2 visite - 21 gen
10 Apr 2007 ... Unlike XHTML2, which doesn't have any support from browser vendors, HTML5 has support from all the major browser vendors except Microsoft. ...
www.digital-web.com/articles/html5_xhtml2_and_the_future_of_the_web/ - 24k -
Copia cache - Pagine simili

X/HTML 5 Versus XHTML 2 ★★⯪☆☆ 3 ratings - [Traduci questa pagina]
So, unlike XHTML 2, X/ HTML 5 is influenced by the current state of the art (Web browser ...
Both X/ HTML 5 and XHTML 2 are at the stage of working drafts. ...
xhtml.com/en/future/x-html-5-versus-xhtml-2/ - 37k - Copia cache - Pagine simili

WG » HTML5, XHTML2, and the Future of the Web ★★⯪☆☆ 1 ratings - [Traduci questa pagina]
11 Apr 2007 ... 3 Responses to "HTML5, XHTML2, and the Future of the Web". Chris Says: April 18th, 2007 at 11:18 am. Nice article! ...
web-graphics.com/2007/04/11/html5-xhtml2-and-the-future-of-the-web/ - 8k -
Copia cache - Pagine simili

(b) YAR rank.

Figure 5: Comparison of web search results without and with relevance values.

4.1.4 System Tuning

The goal of system tuning was to construct a model that given in input data on implicit user feedbacks was able to predict the explicit rate that would be given by the user. To this end, a proper system parameter setting was derived by performing a regression analysis in order to compute optimal weights for the single metrics: PT, RR, and SR. In particular, we accomplished regression analysis on the following five models:

Model 1: $r_1 = \alpha_0 + \alpha_1 \cdot RR$
Model 2: $r_2 = \alpha_0 + \alpha_1 \cdot RR + \alpha_2 \cdot SR$
Model 3: $r_3 = \alpha_0 + \alpha_1 \cdot RR + \alpha_2 \cdot PT$
Model 4: $r_4 = \alpha_0 + \alpha_1 \cdot PT + \alpha_2 \cdot SR$
Model 5: $r_5 = \alpha_0 + \alpha_1 \cdot RR + \alpha_2 \cdot SR + \alpha_3 \cdot PT$

Starting from the user provided explicit rates r_i, the goal here was to derive appropriate values for the constants α_j producing an optimal combination of the metrics RR, SR, and PT to achieve a value close to r_i.

Table 1 presents the results of the regression analysis based on the experiment accomplished by the first group of twenty participants, which produced 650 data records. The adjusted R^2 values show the proportion of variance of the dependent variable, namely the explicit rate of the subjects, with respect to the independent variables, namely the proposed metrics. We can observe that by including all the three metrics (model 5) we gain the maximum amount of variability of the dependent variables with respect to the independent ones. We also observe that the single PT and SR metrics have more impact on the variance than RR metrics. This means that the PT and SR are more strongly related to the explicit rate.

4.1.5 Evaluation Metrics

The quality of the implicit feedback computed by YAR was evaluated by using the data produced in the second round of experiments, in which the second group of six participants was requested to solve three webquests on the topics: economy, politics, and healthcare. Afterwards, we have compared their explicit rates with respect to the implicit ones by means of the Root Mean Squared Error (RMSE).

4.2 Results

The experiments performed by the second group produced a set of 213 data records. Figure 6 shows the number of web pages visited by each subject to solve each webquest. Notice that for the same webquest different subjects visited a highly variable number of pages. Nevertheless, to this end, we can observe that some subjects tend to follow their own trend. As an example, to solve each webquest Subject 1 has visited a number of pages in a restricted range from 6 to 12, whereas Subject 3 has visited few pages except for webquest 1.

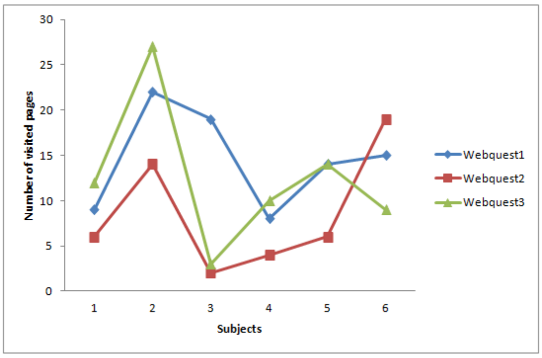


Figure 6: Visited web pages per webquest.

Figure 7 shows the distribution of explicit rates for each subject. This figure allows us to elicit the attitude each subject has exhibited while rating web pages. As an example, Subject 4 has evenly assigned all the different available rate values, whereas Subject 5 has shown less variability by assigning all rates close to 3.

Table 2 shows the RMSE between the implicit feedback predicted through the proposed model and the explicit rate provided by the subjects. We can observe that the combination of all the three metrics

Table 1: Regression results.

Model	adjusted R^2	F-value	p-value	α_0	α_1	α_2	α_3
1	0.375	F(1, 648) = 390.39	<0.001	1.164	0.531	-	-
2	0.571	F(2, 647) = 433.15	<0.001	0.649	0.538	0.291	-
3	0.592	F(2, 647) = 472.66	<0.001	0.570	0.291	0.387	-
4	0.693	F(2, 647) = 732.44	<0.001	0.220	0.422	0.349	-
5	0.857	F(3, 649) = 1293.42	<0.001	-0.126	0.372	0.340	0.337

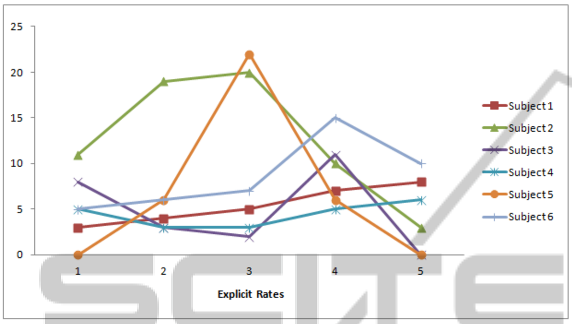


Figure 7: Distribution of rates per subject.

produces the best performances, reducing the error to the minimum average value of 0.286. Moreover, as it occurred in the regression analysis, also here the pairwise combination SR and PT produces errors that better approximate the best RMSE value based on all the three metrics, whereas the RMSE for model 1 and 3 shows that not using SR metrics yields the worst performances. Similar considerations hold by analyzing RMSE for single subjects, except for Subject 1 where the model 2 is worst than model 3.

Table 2: The RMSE values for the analyzed models.

	Subjects						
Model	All	1	2	3	4	5	6
1	0.800	1.109	0.787	0.772	0.762	0.693	1.019
2	0.460	0.719	0.458	0.427	0.447	0.458	0.472
3	0.709	0.688	0.686	0.764	0.771	0.647	0.837
4	0.300	0.410	0.283	0.312	0.490	0.298	0.301
5	0.286	0.283	0.250	0.257	0.339	0.283	0.279

5 RELATED WORK

A well-known strategy to collect data concerning users activities is the *think-out-loud* method (Bath, 1967; Best, 1979; Johnston, 1977; Paul and Rosenkoetter, 1980; McClain, 1983). However, this method is quite invasive, which can significantly influence user's behavior. Further, it is difficult to use it in practice, since it requires considerable efforts in terms of staff personnel to analyze tape-recorded data.

In the scientific literature several other approaches were presented to collect data on user activity (Arroyo et al., 2006; Atterer et al., 2006; Mueller and Lockerd, 2001), especially in the field of web usability studies. However, in these cases the web-site codes need to be modified in order to capture the user interactions, or it is necessary to change the web browser configuration by redirecting all the traffic to an ad-hoc proxy system. All these solutions lack in scalability and cannot be used in large scale experiments, conceived to potentially involve any web users.

One of the first uses of mouse interaction data is in the field of usability studies. Several works exploit user interaction data in order to analyze user behavior and improve usability *Cheese* (Mueller and Lockerd, 2001), *MouseTrack* (Arroyo et al., 2006), *UsaProxy* (Atterer et al., 2006). They all track user activities by logging mouse movements, and produce some visual representation of gathered data highlighting "more interesting" parts of a web page. These data provide useful insights for web designer about the need to rearrange the page layout in order to improve usability.

With the increasing popularity of search engines several relevance measures have been investigated (Kelly and Teevan, 2003; Yanbe et al., 2007). In particular, the increasing number of *Social Bookmarking* systems have suggested that their advantages might be combined with "classic" search tools. A prototype of a system combining PageRank, social bookmarking ranking metrics, and general statistics of user "feelings" is described in (Yanbe et al., 2007). In the same direction, also Google has shown interest for social bookmarking as witnessed by the launch of services like "SearchWiki" and "Google Plus".

However, asking users to explicitly rate web page contents might somehow disturb their activities, which can affect the reliability of the rates they provide. In order to tackle this problem, many approaches have been proposed to implicitly infer user rates. For instance, there are approaches on how to interpret click-through data accurately (Joachims et al., 2005; Jung et al., 2007; Radlinski and Joachims, 2005), or to identify relevant websites using past user activity (Agichtein and Zheng, 2006; Bilenko and White, 2008). Behavioral measures that can be used as evidence of document usefulness include the display time on documents, the number of clicks and scrolling on each content page, the number of visits to each content page, further usage of content pages, time on search result page before first click, and so forth (Agichtein et al., 2006; Claypool et al., 2001; Kelly and Belkin, 2004; Konstan et al., 1997; Xu et al., 2008). Our approach extends these ones by introducing the reading rate metrics, yielding a threefold combination of rating metrics, which has so far proven to sufficiently approximate explicit user ratings. The importance of reading rates is also witnessed by several studies on the different strategies that humans adopt during the process of reading (Hunziker, 2006). By exploiting eye-tracking systems it has been shown that web users adopt peculiar and original reading strategies (Nielsen, 2006; Nielsen, 2008), which differ from those used for printed text.

6 CONCLUSIONS AND FUTURE WORK

We have presented a new model to infer user interests about web page contents from his/her mouse cursor actions, such as scrolling, movement, text selection, and the time s/he spends on the page, while reading web documents. We have embedded the proposed model in the YAR system, a ranking system for the web, which re-ranks the web pages retrieved by a search engine based on the values inferred from the actions of previous visitors. YAR captures mouse cursor actions without spoiling user browsing activities. This is an important issue, because often users are not keen to explicitly rate the usefulness of retrieved web pages, as requested in social bookmarking systems. In order to validate the proposed model, we run several experiments involving a group of twenty-six selected subjects. The results demonstrate that the proposed model is able to predict user feedbacks with an acceptable level of accuracy.

In the future we would like to perform further investigations on how mouse movements relate to user interests in the page contents. For instance, we would like to produce a classification of websites based on typical standard structures (e.g., news sites, blogs, and so on), so as to differentiate the interpretation of mouse movements depending on the type of page being explored. Moreover, we are currently investigating methods to correlate the implicit feedback to the semantics of the original search query, so as to reduce false alarms due to the lexical similarity of words with completely different meanings. Finally, we are planning to exploit user actions to select portions of a web page that could be used as metadata of the pages.

Regarding the experimental evaluation, this is an ongoing process, and there are many issues that should still be faced in the future. First of all, although the results seem to be encouraging, for a complete validation of the proposed model huge experimental data would be necessary. In particular, the system should be used on a large scale in order to track a conspicuous number of user interaction actions for a larger set of web pages. Furthermore, the explicit rank is heavily spoiled by subjectiveness. Thus, the distance between explicit and implicit ranks should not be the unique metrics to measure the effectiveness of an automatic ranking system. For this reason, we are also investigating alternative test criteria involving domain experts rather than naive users in the explicit evaluation of web contents.

Finally, we would like to explore the application of our approach to other application domains, with particular emphasis on usability studies and mashup advising.

REFERENCES

Agichtein, E., Brill, E., Dumais, S., and Ragno, R. (2006). Learning user interaction models for predicting web search result preferences. In *Proceedings of the 29th International ACM Conference on Research and Development in Information Retrieval*, SIGIR'06, pages 3–10, New York, NY, USA. ACM.

Agichtein, E. and Zheng, Z. (2006). Identifying "best bet" web search results by mining past user behavior. In *Proceedings of the 12th ACM Conference on Knowledge Discovery and Data Mining*, KDD'06, pages 902–908, New York, NY, USA. ACM.

Arroyo, E., Selker, T., and Wei, W. (2006). Usability tool for analysis of web designs using mouse tracks. In *Proceedings of Conference on Human Factors in Computing Systems*, CHI'06, pages 484–489, New York, NY, USA. ACM.

Atterer, R., Wnuk, M., and Schmidt, A. (2006). Knowing the user's every move: user activity tracking for website usability evaluation and implicit interaction. In *Proceedings of the 15th International Conference on World Wide Web*, WWW'06, pages 203–212, New York, NY, USA. ACM.

Bath, J. (1967). Answer-changing behaviour on objective examinations. *The Journal of Educational Research*, 1(61):105–107.

Best, J. B. (1979). Item difficulty and answer changing. *Teaching of Psychology*, 6(4):228–240.

Bilenko, M. and White, R. W. (2008). Mining the search trails of surfing crowds: identifying relevant websites from user activity. In *Proceeding of the 17th International Conference on World Wide Web*, WWW'08, pages 51–60, New York, NY, USA. ACM.

Chen, M. C., Anderson, J. R., and Sohn, M. H. (2001). What can a mouse cursor tell us more?: correlation of eye/mouse movements on web browsing. In *Proceedings of Conference on Human Factors in Computing Systems*, CHI'01, pages 281–282, New York, NY, USA. ACM.

Claypool, M., Le, P., Wased, M., and Brown, D. (2001). Implicit interest indicators. In *Proceedings of the 6th International Conference on Intelligent User Interfaces*, IUI'01, pages 33–40, New York, NY, USA. ACM.

Dodge, B. (1995). Webquests: A technique for internet-based learning. *Distance Educator*, 1(2):10–13.

Fox, S., Karnawat, K., Mydland, M., Dumais, S., and White, T. (2005). Evaluating implicit measures to improve web search. *ACM Trans. Inf. Syst.*, 23:147–168.

Golder, S. and Huberman, B. A. (2006). The structure of collaborative tagging systems. *Journal of Information Science*, 32(2):198–208.

Huang, J., White, R. W., and Dumais, S. (2011). No clicks, no problem: using cursor movements to understand and improve search. In *Proceedings of Conference on Human Factors in Computing Systems*, CHI'11, pages 1225–1234, New York, NY, USA. ACM.

Hunziker, H. W. (2006). *Im Auge des Lesers foveale und periphere Wahrnehmung: vom Buchstabieren zur Lesefreude (In the eye of the reader: foveal and peripheral perception - from letter recognition to the joy of reading)*. Transmedia Zurich.

Joachims, T., Granka, L., Pan, B., Hembrooke, H., and Gay, G. (2005). Accurately interpreting clickthrough data as implicit feedback. In *Proceedings of the 28th ACM Conference on Research and Development in Information Retrieval*, SIGIR'05, pages 154–161, New York, NY, USA. ACM.

Johnston, J. J. (1977). Exam taking speed and grades. *Teaching of Psychology*, 4:148–149.

Jung, S., Herlocker, J. L., and Webster, J. (2007). Click data as implicit relevance feedback in web search. *Inf. Process. Manage.*, 43:791–807.

Kelly, D. and Belkin, N. J. (2004). Display time as implicit feedback: understanding task effects. In *Proceedings of the 27th International ACM Conference on Research and Development in Information Retrieval*, SIGIR'04, pages 377–384, New York, NY, USA. ACM.

Kelly, D. and Teevan, J. (2003). Implicit feedback for inferring user preference: A bibliography. *SIGIR Forum*.

Konstan, J. A., Miller, B. N., Maltz, D., Herlocker, J. L., Gordon, L. R., and Riedl, J. (1997). GroupLens: applying collaborative filtering to Usenet news. *Commun. ACM*, 40:77–87.

Mandl, T. (2006). Implementation and evaluation of a quality-based search engine. In *Proceedings of the Seventeenth Conference on Hypertext and Hypermedia*, HYPERTEXT'06, pages 73–84, New York, NY, USA. ACM.

McClain, L. (1983). Behavior during examinations: A comparison of “a”, “c,” and “f” students. *Teaching of Psychology*, 10(2):69–71.

Mueller, F. and Lockerd, A. (2001). Cheese: tracking mouse movement activity on websites, a tool for user modeling. In *Proceedings of Conference on Human Factors in Computing Systems*, CHI'01, pages 279–280, New York, NY, USA. ACM.

Murray, G. (2006). Asynchronous javascript technology and XML (ajax) with the java platform. http://java.sun.com/developer/technicalArticles/J2EE/AJAX/.

Nielsen, J. (2006). F-shaped pattern for reading web content. http://www.useit.com/alertbox/reading_pattern.html.

Nielsen, J. (2008). How little do users read? http://www.useit.com/alertbox/percent-text-read.html.

Page, L., Brin, S., Motwani, R., and Winograd, T. (1999). The pagerank citation ranking: Bringing order to the web. Technical Report 1999-66, Stanford InfoLab.

Paul, C. A. and Rosenkoetter, J. S. (1980). The relationship between the time taken to complete an examination and the test score received. *Teaching of Psychology*, 7:108–109.

Radlinski, F. and Joachims, T. (2005). Query chains: learning to rank from implicit feedback. In *Proceedings of ACM Conference on Knowledge Discovery in Data Mining*, KDD'05, pages 239–248, New York, NY, USA. ACM.

Salton, G. and Buckley, C. (1988). Term-weighting approaches in automatic text retrieval. In *Information Processing and Management*, pages 513–523.

Xu, S., Zhu, Y., Jiang, H., and Lau, F. C. M. (2008). A user-oriented webpage ranking algorithm based on user attention time. In *Proceedings of the 23rd National Conference on Artificial intelligence - Volume 2*, AAAI'08, pages 1255–1260. AAAI Press.

Yanbe, Y., Jatowt, A., Nakamura, S., and Tanaka, K. (2007). Can social bookmarking enhance search in the web? In *Proceedings of the 7th ACM/IEEE Joint Conference on Digital Libraries*, JCDL'07, pages 107–116, New York, NY, USA. ACM.