Profiling the user: A ringside view

Bedekar M.V.* and Deshpande Bharat
*CS-IS Group, BITS-Pilani, K. K. Birla, Goa Campus, Zuarinagar, Goa, 403 726, India,
mangesh.bedekar@gmail.com, bharatmsu@yahoo.com

Abstract: We all regularly use the internet for a variety of reasons. We do like some sites and dislike others. There can be various reasons for liking and disliking sites. Some sites interest us, some sites are visited by us often, some are visited periodically, some simple annoy. Our Internet usage is also pretty much the same everyday barring weekends. A average user logs on to the internet at nearly same times everyday, surfs some sites regularly some new sites at times and perform repeated action on sites, more or less. The browser is used as an intermediatery for developing a system which identifies these usage patterns, learns them and then uses it to enhance and personalize our surfing behavior. The system is smart enough to prefetch the right pages at the right time and display them in the browser for the user, all without any manual intervention.

Keywords: User Profile, Personalization, Customization, Prefetching

I INTRODUCTION
Let’s start with a real life scenario. Mr. Kumar comes to work every weekday at 9.00 am. He logs on his computer and checks office mails. Then he uses mail to check his personal mails. He then goes on to check the days’ opening Stock index, then to his favourite sports site for all the days’ sports news. Before leaving office, at 5.00pm, Mr. Kumar again checks his personal mail account, the days’ closing stock value and the TV schedule page of his favourite channel for the evening. There is a certain pattern to this behavior, at least on week-days. It is these patterns which have to be identified based on the users’ internet usage. The browser is used as an intermediatery for developing a system which identifies these usage patterns, learns them and then uses it to enhance and personalize our surfing behavior. The problem at hand is first to profile the user, then as per the profile, customizing the browser to display certain sites at certain times. Profiling the user tries to create a persona of the user whereby it records the sites visited by the user, the time of the day, the day of the week, the frequency of returning to the same sites, etc. These small bits of information gathered from the browser are stored in a systematic fashion. This information is then used to present the user with a more comfortable browsing experience by displaying the sites he visits more often, at the right times, prefetching the links so as to speed up the process. All in all it gives proactive feel to the user, which is tailored as per his profile of usage [4].

Explicit profiles have to be updated by the user specifically from time to time for them to be equally relevant across time.

These profiles can be created in the following ways.
Ask questions
Ask the user to fill forms
Get relevance feedback from the user
Generate a interest list for the user
Update interests time to time

IMPLICIT Profile Creation
For a repeat user, profiling implicitly makes sense as we have some past performance history. Also, looking at the browse entries a lot of things can be learned about the user. User’s bookmark list in the browser is an indication of his interests. Users past browsing patterns over a period of time, with URL’s visited, order of there visiting, indicates interests and changes to users interests. Users past search queries (history) on search engines provide important hints on the user’s future interests / topics of further interests. Out of the URL’s clicked by the user, the time spent on each URL-Page and subsequent pages linking out from the particular page can be found. The domain to which these URL’s belong can be identified as more relevant to the user. The time spent by the user scrolling a page indicates its relevancy. The frequency of key words in the saved/printed documents can be identified. The (TFIDF) Term Frequency Inverse Document Frequency algorithm determines the relative importance of this page to the user. If the user downloads a page / prints a page / bookmarks a page, it confirms its relevance. All the above mentioned factors are retrieved with there respective Date and Time stamps so as to pinpoint timely relevancy of topics/pages/sites.

II CATEGORIES OF PROFILE CREATION

EXPLICIT Profile Creation
For a first time user of a particular site –OR– For a new site, newly created on the net, The explicit method is most relevant as nothing is known of the user’s interaction with the new site, or the content, structure of the new site. A first time user can be asked for feedback about the site or satisfaction of getting the required content on the site, but the user may hardly fill the same.

UPDATE interests time to time
Generate a interest list for the user
Get relevance feedback from the user
Ask the user to fill forms
Explicit profiles have to be updated by the user specifically from time to time for them to be equally relevant across time.

These profiles can be created in the following ways.
Ask questions
Ask the user to fill forms
Get relevance feedback from the user
Generate a interest list for the user
Update interests time to time

IMPLICIT Profile Creation
For a repeat user, profiling implicitly makes sense as we have some past performance history. Also, looking at the browse entries a lot of things can be learned about the user. User’s bookmark list in the browser is an indication of his interests. Users past browsing patterns over a period of time, with URL’s visited, order of there visiting, indicates interests and changes to users interests. Users past search queries (history) on search engines provide important hints on the user’s future interests / topics of further interests. Out of the URL’s clicked by the user, the time spent on each URL-Page and subsequent pages linking out from the particular page can be found. The domain to which these URL’s belong can be identified as more relevant to the user. The time spent by the user scrolling a page indicates its relevancy. The frequency of key words in the saved/printed documents can be identified. The (TFIDF) Term Frequency Inverse Document Frequency algorithm determines the relative importance of this page to the user. If the user downloads a page / prints a page / bookmarks a page, it confirms its relevance. All the above mentioned factors are retrieved with there respective Date and Time stamps so as to pinpoint timely relevancy of topics/pages/sites.
These profiles can be created in the following ways,
Identify pages / sites visited often
Identify amount of time spent on pages / sites
Identify scrolls, mouse clicks on pages / site
Identify pages / sites, where user spends most time
Identify pages / sites bookmarked
Identify pages / sites saved

Comparing IMPLICIT & EXPLICIT Profiles
IMPLICIT profiles are,
Non-intrusive
Automatically updatable
Can be assumed to be indicative after some time
Difficult to implement

EXPLICIT profiles are,
Intrusive
Have to be manually updated
Are assumed to be indicative right from the start
Easy to create, difficult to populate
Easy to implement

Combining IMPLICIT and EXPLICIT Profile Creation [HYBRID Profile]
The advantages of both can be clubbed together to achieve a more powerful profile methodology creation and use.

III INFORMATION GATHERING
Some personal information is gathered from the user for the profiles are,
Age range
Gender
Marital status
Highest Educational qualification
Profession
Income range
Mother tongue language
Language preferences in order
Geographic location
Continent / Country / Region of location
Approximate address / Area of residence

The above mentioned information maintains the anonymity at the same time conveying a lot of important information for the profile. Interests of people in the younger age group tend to shift between topics faster than people of the elder age group. People of Higher technical qualifications demand more minute details on topics as compared to people of a lower qualification. People of Lower technical qualifications demand more general details on topics as compared to people of a higher qualification. A professional biologist may be interested in more biological content in pages than personnel from economics and vice-versa. A student look for questions and solutions of questions on focused topics, and more often, than a professional who looks for more related and advanced content on the same topic. While shopping for costly / luxury products, the income range of the user influences the choices of products he is interested in getting details of / buying them. Languages known by the user to read/ write greatly influences the sites the user may visit or not visit. Country and Culture of that place also plays a vital point in the user’s decision making process. Address/ Area of residence convey a lot of information of the user and his interests.

IV PROFILE STRUCTURE & CONTENT
The profile is created at various levels using a variety of data structures and data storage mechanisms. Information collectors/ holders are created with varying weights.

Profile Creation and Updates
The user’s on-line behavior is continuously tracked.
The profiles are created off-line, after the users surfing session ends.
The profiles are accessed on-line during surfing.
All subsequent updates are done off-line.

Types of Profiles
Term based / Item based profile
Category based profile
Link / Structure based profile[2]

Term Based / Item Based
They include the Items / terms the user is looking for / may look for / already known are stored and these terms are weighted as a ratio of frequency of occurrence, the time of occurrence.

Category based profile
They include the Categories and sub-categories organized in hierarchies.
All the keywords (as mentioned above) are categorized in a hierarchical manner and aggregated differently. Each category gets a weight depending on the overall weights of all the terms contained in it.

Link / Structure based
They include the Pages / Sites / Domains visited by the user and the inter-connection between them. Weights are attached as a ratio of frequency of occurrence, the time of occurrence. Google’s PageRank is a good algorithm to decide on the PR of each link, then to sort them in decreasing order, in order to decide on the relevance.

IV PROFILES AND WEIGHTS
The user profiles change with time. The more frequent the usage the more the sites visited, the more the changes. Also as per the sociology standpoint the users’ interests change over time. This forces us to attach varying weights to all items of the profile or aggregate profiles into various categories to suit our purpose.
As the users' interests in particular keywords increase, as is evident by the frequent occurrence of these new words in the profile, a positive weight is attached to the term. [6]

Profiles can also be weighed on,
- Short – Term
- Medium – Term
- Long – Term

**Short – Term**
All the ‘Term based / Item based’ profile elements are used here with the ones which are relatively newly last accessed, say a week to a fortnight.

**Medium – Term**
All the ‘Term based / Item based’ profile elements, ‘Category based profile’ are used here with the ones which are accessed over a decent period of time, say a fortnight to a few months.

**Long – Term**
All the ‘Term based / Item based, Category based and Link/ Structure based’ elements are sued here, which can span backwards for as long as the data structure can support, or, as long as the user wishes his data to be stored, typically it spans a duration of around a couple of years. As time progresses, and as users interests change, new terms are added to the Short-Term Profiles. They stay there till a particular minimum threshold of overall weight is not met. As the weight crosses the threshold, on the minimum side, the terms shift to Medium-Term Profiles, then to Long-Term Profiles as the users interests in particular items wanes. The keywords as permanently stored in the Long-Term Profiles. They are never deleted by the system. The Exception being, the user, at any time, may always delete the terms from any of the profiles associated with the user.

Profiles can also be created as,
- Individual
- Collaborative

Individual profiles are very private in the sense only the users’ access behavior is used and stored [1]. Collaborative sharing can start if the profile is stored on the proxy server. Here along with individual profiles groups of profiles are created with similar patterns. It can also occur if the user willingly shares his individual profile with another user / group of users [5].

**V PROFILE STORAGE AND ACCESS**
All the profiles thus created are stored in secure containers and accessed when required. The profiles can be compressed and stored to save space but this slows down the system.

The location of storing the profile is also as important as creating it.

The profiles can be stored at,
- Client Side
- Proxy Machine
- Server side
- Institution / Organisation / Company Server

Each of these implementations has their own characteristics as explained below.

**Client Side Storage**
Storing the profile on the client’s machine makes it safe, fast and secure.
It reduces the computational, communicational and storage cost of elsewhere.
All the end level personalisation is done on the client machine using this profile in the fastest way, as compared to any other implementations.
Updates to this profile can be affected immediately.
Regular, daily, continuous updates are possible.

**Proxy Machine Storage**
Storing the profile on the proxy machine enables collaborative sharing.
The efforts of one user can help other users attached to the same proxy server.
Re-working is reduced to a huge extent of collaboration is enabled as it saves on computation and communication time.
Updates to this profile can be affected a lot easily.
Daily, continuous updates are possible.
Example The Squid Proxy Server, along with the Profile Server.

**Server Side Storage**
The advantage here is that the profile becomes portable and can be accessed from any place, where the server is reachable.
The drawback is many users having profiles stored a single point makes it a little un-safe, the speed of the machine reduces and security is of vital importance here.
Updates to these profiles cannot be affected immediately.
Regular updates are not possible, only periodic updates can be permitted, depending on the number of users and load on the server.
Example the Web Server along with a Profile server.

**Organisation Server Storage**
Storing the profile on these server machines enables collaborative sharing, as employees of the same company have the same business interests. The efforts of one user can help other users attached to the same proxy server. Re-working is reduced to a huge extent of collaboration is enabled as it saves on computation and communication time. Updates to this profile can be affected a lot easily. Daily,
continuous updates are possible. Example a Proxy Server or Web Server and a Profile server

VI ISSUES with Location of Profile storage

User Profiles have to be very quick in responding to user’s requests as well as private to him. Profiles should be readily accessible but at the same time be safe and secure too. Hosting the profile at a location can have its own issues. Some of these issues are as follows.

Issues with Server side Profile storage –
Frequency of updates to profile depends on internet speed and quality of connectivity.
Usually implemented as a separate system
Lots of pre-processing is required before unique usage patterns can be identified.
General user models can be created.
Even the servers company has access to revealed personal information at all times.
Can understand context of user

Issues with Client side Profile storage –
How large Profile is to be created / maintained vis-à-vis the benefits got out of it.
Usually implemented in conjunction with the browser itself
Active time spent on a page (dwell time) becomes an important issue.
Specific models of the same user can be created.
Only user has access to full personal information.
Can be completely personalized as per each user’s individual traits

Issues with Proxy side Profile storage –
Size of Profiles is a issue when the number of users increases.
Reducing response time to users becomes an important issue.
This intermediate server has access to a lot of personal information of many users.

VII IMPLEMENTATION DETAILS

The current implementation stores the profile on the client site and is working for Short-Term profiling of the user. It has been coded on open source systems. It works on the WAMP Architecture [3]. The browser Mozilla Firefox installed on the client machine talks to Apache on the client machine as a proxy before connecting to the internet. The Mozilla Firefox browser has Grease Monkey installed as an extension for running all the scripts on web pages accessed by the user at run time. The PHP scripts run off-line computing the new weights after the users’ browsing session has ended. Google desktop is also used in conjunction for accessing files stored on the machine so they are not saved, downloaded again.

VIII PROFILES AND PRIVACY

When everything that the user accesses is logged, there is a concern of privacy coming in. The user is given complete access to his profile, so he can remove any entries in the profile at will. The tradeoff is more the entries in the profile the better can it can produce results and vice-versa. The user may wish to enable or disable sharing his profiles when located at proxy/ server side/ institutional servers. The Term based/ Item based profiles will be totally private to the user. For Category based profile, only the Categories are shared, the content is not shared. For Link / Structure based profile, the sharing will go from the abstract to the precise in the order of Domains -> Sites -> Pages -> Links, if enabled by the user.

IX COMMENTS

Initial results are promising. The implementation is currently being fine tuned to meet the exact needs of the user. The system can proactively load a page on the browser at the determined time, if kept open.
If the browser is opened only at the determined time, the page loading takes some time as the database needs to be accessed and action taken. For Slideshows from websites, frequently seen by the user, the slide pages are also loaded in different tabs automatically. All domains, hosts accessed by the user get logged with the date and time stamp of accession, to the last second.

X CONCLUSION

A conclusion section is not required. Although a conclusion may review the main points of the paper, do not replicate the abstract as the conclusion. A conclusion might elaborate on the importance of the work or suggest applications and extensions.

XI REFERENCES
