Journal of Universal Computer Science, vol. 17, no. 13 (2011), 1830-1853
submitted: 4/1/11, accepted: 27/7/11, appeared: 1/9/11 © J.UCS

An Inquiry into the Utilization of Behavior of Users in Personalized Web

Michal Holub, Mária Bieliková
(Slovak University of Technology in Bratislava
Faculty of Informatics and Information Technologies
Ilkovičova 3, 842 16 Bratislava, Slovakia
{holub, bielik}@fiit.stuba.sk)

Abstract: Nowadays we see successive transformation of the Web into its personalized form. In order to personalize the content to suit each user's requirements we need to acquire the user's interests. Utilization of implicit feedback is the most suitable and unobtrusive way of doing so. In this paper we present various forms of implicit feedback and their application in the estimation of user's interests. We propose a method of link recommendation based on the recorded actions users take while visiting a website. We employ collaborative filtering to predict user interest to unvisited pages. We present an evaluation of our method using the web portal of our faculty where personalized recommendation of links to interesting events is provided for visitors.

Key Words: implicit feedback, recommendation, user actions, navigational patterns, interest estimation, personal calendar
Category: H.3.3, H.5.4

1 Introduction

Everyone has a characteristic behavior which also reflects our actions on the Web. People belonging to similar social groups or with common interests share also some behavioral features. It is presumable that these people browse the Web in similar manner and take similar actions while interacting with it. As we do not know the social relationships of users from the reality we recreate them using clustering based on the behavior of users while interacting with a website [Zehraoui et al. 2010, Zhu 2010]. The resulting groups and patterns are used to make recommendations and to personalize the Web for each user [Xie and Phoha 2001, Mobasher 2007]. We suppose that users in the same group, who share similarities in the way they interact with websites, have some common interests or play similar roles in the reality.

The behavior of users on the Web is characterized by the patterns found in their navigation on websites [Cadez et al. 2003, Dalamagas et al. 2007]. Moreover, the users show their interest in a particular web page by providing feedback on it. The feedback can be either explicit or implicit.

User's feedback is being used in recommendation and personalization tasks, which utilize a user model constructed from his behavior [Ahmed et al. 2009]. From the provided feedback the interest of a user in particular item (the main entity which the web page refers to, e.g. a movie, music band, news article, or the web page itself) is computed. Collaborative filtering is being widely used for the prediction of user's interest in unvisited items or web pages [Resnick et al. 1994, Sarwar et al. 2001].

The problem is how to determine whether a user found the visited web page interesting. In this paper we deal with a transformation of the implicit feedback into the estimation of user's interest. We track actions like time spent on a web page, scrolling and text copying actions, which form a base for the estimation of user's interest in particular web page. We use navigational patterns to group users with estimated similar interests and recommend links among them. The recommended links point to announcements about upcoming events extracted from a selected web portal. Employing the collaborative filtering we predict user's interest in those events. As a result we create for each visitor his personalized calendar, which consists of events we estimate as interesting for the visitor.

We mainly focus on hyperlinks because following them is the most commonly used pattern for information access on the Web [Cockburn and McKenzie 2001, Tauscher and Greenberg 1997]. It is done in more than 50 % of the cases (other include typing a URL or selecting a link from the browsing history). Therefore the biggest improvement of user's browsing experience can result from personalization of displayed links and their recommendation among similar users.

The rest of this paper is organized as follows. In Section 2 we present related work done in the field of feedback collection for the purpose of interest estimation. We also present approaches to description of navigational patterns of web surfers. In Section 3 we describe our proposed method for adaptive navigation support using automatic interest estimation and navigational patterns discovery. Evaluation of the proposed method involving experiments with the web portal of our faculty is presented the Section 4. Finally, Section 5 concludes our paper with discussion of issues and directions for the next research.

2 Related work

Utilization of user's behavior for personalization of the Web is currently an active research area. We consider two aspects of user's behavior while interacting with a website:

- The behavior on a single web page represented by the actions done by the user.
- The behavior on the whole web portal expressed by the sequences of followed hyperlinks.

We present related work in these areas. For the behavior analysis we consider a web page to be a set of elements. Individual elements are mainly hypertext links and parts of the web page containing web objects (a paragraph about an upcoming event, main content of a news article, etc.). During the interaction with the page's elements the user takes various actions. In order to personalize the content of web pages according to the user's behavior we need to track and record actions the user takes. We analyze these records and utilize them in a user model. Users are grouped together according to their behavior—users with similar behavior are put into the same group. Further, we can track and analyze the behavior of the whole group. The results are used for recommendations of the content among the group members.

2.1 Feedback for user modeling

The feedback a user gives after visiting a web page or seeing an item is the best indicator of his preferences. Each recommender system needs to create a user model in order to provide recommendations for single users. This model needs to be constantly updated in order to reflect the changes in user's abilities and needs. Updating the user model would be impossible without any kind of feedback provided by the user after he was presented some information.

There are two forms of feedback the user can provide:

- *Explicit feedback*, which is provided intentionally by the user, often as a response to a query (e.g. opinion about an item). The query can be in the form of a single choice question (e.g. the web page was or was not interesting), scalar rating (e.g. how many stars the user awards to a movie) or more occasionally a text answer. The disadvantage of explicit feedback is that it requires non-zero user effort and the users may be therefore unwilling to provide it.

- *Implicit feedback*, which does not require the user to exert any extra effort. The user interacts with the web page in order to fulfill his information need as usual. The feedback is gathered implicitly from the actions he takes (e.g. on which link he clicks) and from the context (e.g. which links were visible to him). This has an advantage that we do not bother the user.

Explicit feedback is the easiest way of data collection for modeling of users. Before the user starts interacting with a web portal we can ask him to fill in an electronic questionnaire. It can contain keywords regarding his interests or current information needs and goals. Using this type of explicit feedback it is possible to collect any kind of information from the user. As far as the user answers the questions truthfully, the information collected is exact.

The problem with this approach is that the users are in general unwilling to fill in any kind of forms before they visit a certain web page. It consumes time and without proper motivation the users will not do it. Further problem is that the users can have different sensitivity to ratings. When asking about their skills users can overrate or underrate themselves.

Jawaheer et al. [Jawaheer et al. 2010] present a study of how users provide explicit feedback in an online music service, and how this feedback can be used in a music recommender system. They analyzed ratings of music tracks on Last.fm, where people can explicitly mark a song as *loved*. This influences the list of recommended songs which they are later provided. The result of the study is that users provide this kind of feedback very rarely. Also, its volume decreases over time as the initial enthusiasm of the users about the system fades away.

Implicit feedback, on the contrary, needs to be analyzed in order to better understand its meaning. Velayathan and Yamada [Velayathan and Yamada 2007] used monitoring of user's behavior in order to estimate his interest. The behavior on web pages visited by the user was used as an implicit feedback determining his interest in that page. Using augmented web browser they recorded all actions the users had taken. They found out that various actions are indicators of either positive or negative interest. Actions with positive meaning are e.g. printing the web page, copying text into clipboard or bookmarking the page. When the user closed the browser window (or tab) before the web page was fully loaded or when he spent small amount of time on it, it was considered to be a sign of negative interest.

Additionally, implicit feedback can express both positive and negative interest. To make matters worse, one particular action can have different meanings in different contexts or domains. The most common way of getting implicit feedback is to record on which hyperlinks the user clicks. When he reads a piece of text and then clicks on a link related to it, it can be seen as a sign of positive interest. Conversely, when he does not click it can be seen as a sign of negative interest. However, this is not always true. In general, we are unable to tell if he did not click on a link because he is not interested in the subsequent web page, or simply because he did not notice the link. Likewise, in general we cannot tell the interest of a user from clicking on a link. The link could have had for example a misleading caption.

Clicking on a link can determine user's interest in the content with annotated links. An example would be a web page of news provider. Usually, links to articles are formed by their titles. They also contain short introduction or summary of the news article. If the user wants to read the whole article he can click on the link to see it. In such situation clicking on the link can be considered as a sign of positive interest [Das et al. 2007, Suchal and Návrat 2010].

Monitoring user's clicks is often used for link recommendation. Baraglia et al. [Baraglia et al. 2006] recorded all links on which the users have clicked during a browsing session. They used clustering for grouping of links which are often used together. Every link was put into exactly one cluster. The links from the cluster with largest overlap with the current session of a user were proposed as the recommendation. These links are the ones that were frequently used by other users in the same context.

Another good implicit feedback indicator is the time the user spends reading a web page [Claypool et al. 2001, Fox et al. 2005]. An example here is again the domain of news articles. Morita and Shinoda [Morita and Shinoda 1994] showed that when the user is interested in particular news article he spends more time reading it than he would without being interested in it. It is also very probable that when the user likes an article, he will read related articles as well.

Explicit feedback is usually present in the form of item ratings [Hofmann 2004, Rennie and Srebro 2005, Zhou et al. 2008, Pilászy and Tikk 2009], which could be e.g. movie ratings on video portals. Users obviously do not rate all items, therefore collaborative filtering is used for ratings prediction in recommendation tasks [Goldberg et al. 1992].

Explicit ratings represent explicit user feedback, which is difficult to obtain. In most cases personalized systems use only positive implicit feedback [Gauch et al. 2007]. Recently emerged *one-class collaborative filtering* methods can be used to predict user's interest based on this feedback [Hu et al. 2008, Pan and Scholz 2009, Li et al. 2010]. There are also approaches which use both explicit and implicit feedback at the same time. Liu et al. [Liu et al. 2010] presented an approach in which they combined both types of feedback with different weights. They proposed a model for integration of various scales used in different types of feedback. This resulted in an improvement of the recommendation quality.

Lee and Brusilovsky [Lee and Brusilovsky 2009] proposed slightly different approach utilizing negative feedback. Their study done on a system recommending job offers shows that negative feedback helps to distinguish between good jobs and bad jobs. They considered closing a job offer without any further action as a negative interest indicator arguing that the job offer did not attract the user.

For the recommendation and personalization purposes users are as a rule grouped based on similarities in their behavior. This helps when a new user visits the website. At first, we do not have any information about his interests. Knowing the group he belongs to serves for guiding him through the web portal so that he is not confused with the information and hyperlink overload. Users who already visited the web portal left digital footprints on the links they used. These are used for recommendation of frequently followed paths for the new users [Brusilovsky 1996]. Returning visitors also benefit from this because it is very probable that they will need information which can be found following one of the most commonly used sequence of links.

Although people belong to different groups and have different information needs, most web portals provide all of them with the same content. Visitors often see information in which they are not interested, which the web portal presents to attract a wide range of users [Barla et al. 2009]. The answer to this problem lays in personalization of the website's content. Sheng et al. [Sheng et al. 2008] presume that the interest of visitors also depends on the geographic location of the visitors. They discover geographical-specific interest patterns in the web click data.

Implicit feedback is more suitable for the automatic interest estimation as it does not require any further actions from the user. Some interest indicators proposed are domain specific, e.g. the negative feedback on the job offers web portal. We are looking for a solution which would work on various different web portals; therefore it should be domain independent.

2.2 Recording user's actions

There are several places where the web usage actions can be recorded. The selection of the place determines the type of actions we record. It also influences how precisely we are able to describe user's behavior. The places for actions recording are [Hu and Zhong 2005]:

- *Server side* action recording – this kind of action recording is the easiest one. It can be done by the webmaster. All HTTP requests from the clients are stored in a log files. The disadvantage is that we cannot distinguish among individual users, as there can be more users behind one IP address. We are also unable to record the interaction of the user with the web page (actions done using mouse and keyboard).

- Recording actions on a *proxy server* – here, actions are recorded between the browser of the client and the web server. It is useful for monitoring the behavior of groups of users using the same proxy server. Moreover, if the proxy server requires authentication we can distinguish among the users. Again, the disadvantage is that the user needs to approve this kind of monitoring and set up the browser to use the proxy server.

- *Client side* action recording – this data can reveal the complete interaction with a web page. We can also distinguish among different users. However, this is the most difficult option to achieve as the user needs to approve this kind of monitoring.

- *Application layer* action recording – this monitoring is done by the application itself. It can contain the most detailed information about the actions. On the other hand, each application needs to implement it. We also need to ensure reliable transfer of recorded data from the client to the server.

Application level recording of actions was used in [Andrejko et al. 2006]. The authors used a web service which collects data from various client tools recording actions done in different applications. There was a special recorder for each application monitored. The advantage is that events and actions from various applications are collected at one place where they can be analyzed. The data is enriched with semantics and the state of the application at the time when the action was taken. The events are described using an ontology, which defines their attributes and relations.

Similar approach was used in [Krištofič and Bieliková 2005]. Adaptive systems used by clients were monitored by means of wrappers. Each system we want to monitor requires a special wrapper. The advantage is that the wrappers transform the recorded data into a standardized format which serves as an input for content personalization and recommendation method.

The monitoring of user's actions on the client side can be done in various ways. One of the easiest way is to use cookies in which data can be stored. The second option is to create a browser add-on which can record the interaction between the user and the browser. The advantage is that we also have the context in which the web page is used (e.g. what the user stored in his bookmarks or if the user uses tabs for browsing). The disadvantage is that we need to create different add-ons for different browsers. Each browser can record different types of actions which introduces ambiguity.

Another option is to use a script added to a web page which monitors the actions taken by the user. It is browser independent and can record general actions like interactions using the mouse. Special features provided by the browser are usually impossible to record using this approach.

User's actions can also be monitored using special toolbars which can be installed to the browser [Manber et al. 2000]. The toolbars are developed mainly by large companies in order to make the interaction with company's services easier. Toolbar can also be used to record actions done on a certain web page. The disadvantage is that we need to persuade users to install special toolbars into their browsers.

Server side user monitoring is easy to realize, but it is very noisy and cannot capture a lot of interactions with the web page. On the other hand, client side (together with application layer) user monitoring can capture various actions, but is very intrusive and often requires installation of additional components. From this perspective the most suitable solution for our needs is monitoring user's actions via a proxy server. The proxy server can insert an action recording script to each web page (without modifying its content) which will send recorded actions to the server. The user does not need to install additional items; the only thing he has to do is to set up his browser to use the proxy server. This has some privacy issues which can be minimized by using anonymous user identifiers, letting users see what the system logs about them, and publicly providing source code of the script. The motivation for users to use a proxy server is in the added value provided by personalization of web pages and recommendation of interesting items.

2.3 Navigational patterns

Apart from implicit and explicit feedback preferences of a user are also reflected in the way he navigates in the website. This includes the links he uses and their sequences, as well as the order in which he uses them. Some sequences may be used more commonly than others. If there is a subsequence, which is often repeated in many sequences, we generalize it to a pattern. When we discover the context in which this pattern is used, we group the users whose browsing sessions contain this pattern and assign them the same context. These users share some common characteristics or interests and we use it for link recommendation.

Web usage mining is being used for the analysis of the sequences of followed links. The results are used for detecting common attributes of behavior of users and for adaptation of the web portal for this group of users [Ting et al. 2005]. Likewise, we can employ web object usage as a pattern for finding similar users [Niemann et al. 2010].

Web usage mining is usually done in three steps [Baraglia and Silvestri 2007]:

- Data preprocessing – in this phase sequences of links are mined from the access logs stored on the web server.
- Pattern detection – in the sequences of links patterns are detected using e.g. association rules or statistical analysis.
- Pattern analysis – discovered patterns are being evaluated and the user model is being updated.

In the data preprocessing phase we need to distinguish among different sessions of users. Then we combine the links used in one session into a sequence ordered by the time each link was used. Then we detect patterns in the sequences. We can either discover predefined patterns or new patterns composed of often repeating sub-sequences of links. In the pattern analysis phase we determine the context from discovered patterns; the user is being assigned a label according to the dominant pattern(s).

There are four elementary patterns which can be found in the sequences of items [Canter et al. 1985]:

- *Path* – a sequence in which nodes do not repeat.
- *Ring* – a sequence that starts and ends in the same node.

- *Loop* – a sequence that goes through already visited node.
- *Spike* – a sequence that goes back through the same trail.

We can also discover these patterns in the sequences of links that the users followed on a website. The patterns reflect the purpose of the user when browsing through a web portal. Their analysis reveals whether the user is new to the portal and is looking around, or whether he is regular visitor and knows exactly where to find the desired information.

In addition to the elementary patterns, the whole sequences of links may resemble one of these complex patterns [Clark et al. 2006]:

- *Stairs* – describes the situation in which the user immerses into the web portal without returning to the previous page. This is a common behavior when inspecting a new web portal.
- *Fingers* – describes the situation when the user follows a link to a certain web page and then goes back to the previous one. This behavior is common when we are trying to find something particular but we do not remember on which page it was. We try every link in order until we find it.
- *Mountain* – describes the situation when the user visits a series of links and then he goes back. It is the combination of previous two patterns.

The complex patterns are difficult to detect automatically. Moreover, they do not provide as much space for user's behavior analysis as the elementary patterns. Therefore, we detect the elementary patterns in the sequences and use them to group users with similar browsing behavior.

3 Navigation support based on behavior analysis

We propose a method for adaptive navigation support which is based on the analysis of user's behavior while surfing through a web portal. In general, we record actions which are positive or negative interest indicators. We compare their values with the values of other users who visited the same page. Afterwards, we compute user's interest in each visited web page. For the prediction of interest on unvisited pages we use collaborative filtering method. We consider here the users with similar patterns in their browsing sessions. The predicted interest is associated with web objects extracted from a selected web portal, which are then recommended to the users with high predicted interest.

3.1 Automatic interest estimation

We determine the interest of a user in every visited web page within a website. This process is based on the analysis of actions he takes while interacting with web pages of the portal. Actions determine short-term characteristics of the user, which can frequently change.

3.1.1 Interest indicators

Users take various actions while reading a certain web page. These actions reflect the degree of user's interest in that page. Therefore, we can use them as the implicit feedback. This has the advantage that the user does not need to take any further action to rate the web page. On the other hand, we have to interpret the sequence of actions to estimate user's interest. This is not an easy task, which is influenced also by the fact that some actions can have positive or negative meaning depending on the context, and having just implicit feedback, it is not straightforward to determine their polarity.

We consider actions taken on a web page to be the interest indicators. According to their meaning we divide interest indicators into three categories:

1. positive interest indicators,
2. negative interest indicators, and
3. context-dependent interest indicators.

When we detect an action with assigned positive meaning we increase the current value of the user's interest. Conversely, when we detect an action with assigned negative meaning we decrease the current value of user's interest. This way we evaluate all actions a user does on a web page in a single session. We then associate the final value of his interest with particular web page and the session.

Context-dependent interest indicators can have both positive and negative meaning. To determine the current meaning we analyze the social context, which is the behavior of other user's who visited the same web page in the past. We consider a context-dependent interest indicator to have:

– negative meaning, if its value is lower than the average value of this indicator from other users, and

– positive meaning, if its value is higher than the average value of this indicator from other users.

Time spent on a web page is good example of context-dependent interest indicator. Consider that a user spends 1 minute reading certain web page. To determine, if this is a sign of positive or negative interest, we compute the average time spent on that page by similar users. If the average time is higher than 1 minute, current user's interest is negative; otherwise it is positive.

We record the actions done on a web page only when the user is active, i.e. when he interacts with the web page via mouse or keyboard. The recording is done according to Algorithm 1. These steps are repeated periodically while the web page is displayed.

Algorithm 1 Recording user's actions

```
user active := false
if action $a$ occurred then
    indicator a += 1
end if
if mouse moved event or scroll event then
    user active := true
end if
if user active = true then
    save all indicators
    user active := false
    reset all indicators
end if
```

Using this algorithm we monitor the user only when he is active. We use this to measure the time spent on a web page. If we measured the total time from loading the web page until unloading it (e.g. closing the browser's window, loading other page), we could get inaccurate results. The user might leave the computer for a while and this time would be included in the total time spent. So we measure only the time the user actually spends interacting with a web page. This gives more accurate results, even though for some users there are sometimes situations when the user reads a page without any interaction with it.

3.1.2 Analysis of actions taken

In order to estimate user's interest in a (visited) web page we analyze the actions (interest indicators) which he takes on that particular page. The more positive actions a user takes, the more he is interested in the page, and vice versa. Subsequently, we use the information about the user's interest to:

- predict his interest in unvisited pages, and
- recommend interesting pages to him.

Since there are interest indicators with context-dependent meaning, we divided their possible values into three intervals:

- above-average value, IF $value > average + K$ %

– average value, IF $value \in\ [average - K\ \%\ ;\ average + K\ \%]$

– below-average value, IF $value < average - K\ \%$

The selection of interest indicators to record depends on the target domain and platform used. One of the goals of our work was to come up with a solution which would work with all modern web browsers. For this reason we are not able to record some special actions, like adding a web page to bookmarks, without implementing an add-in for every browser.

Proposed method is general and works with any combination of interest indicators. For our experiments we decided to record these three interest indicators:

– *time* actively spent on a web page,

– number of *scrolling* events occurred, and

– *copying* a text into clipboard.

These actions can be recorded independently from the used web browser or platform. In our case, the time spent on the web page is not absolute; it is the number of intervals in which the user was actively interacting with the web page. The exact time depends on how often we check user's activity. In our experiments we did checks of user's activity every 5 seconds.

We express user's interest as a value [0; 1] with 0 meaning no interest in visited web page and 1 meaning absolute interest in visited web page. After detecting an action with positive meaning we increase the value of user's interest by 0.1. We transform the context-dependent actions according to the scheme shown in Figure 1.

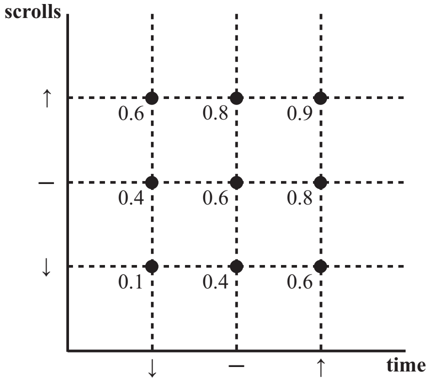


Figure 1: Transformation of context-dependent actions to interest of a user: ↑ means above average, ↓ means below average.

We compute user's interest for every page he visits. Afterwards, we associate the value of interest with each web page. Alternatively, we may analyze the web page, extract web objects from it and associate the interest of a user directly with the objects. Web objects are parts of the web page referring about a certain product, the main content of a news article, a paragraph about an upcoming event, etc.

3.1.3 Interest prediction and link recommendation

A user usually visits only a small subset of pages that are part of a web portal. As mentioned in the previous section, we use the actions he takes to compute his interest in each page of this subset. For other (not yet visited) pages we predict his interest using collaborative filtering inspired by [Sugiyama et al. 2004]. Collaborative filtering operates with items and their ratings. It is commonly used to predict user's ratings of items according to how similar users rate the items, so it takes the social context into account .

We use collaborative filtering with our data about user's interest. The web pages are the items to be rated. In our case, the rating is the value of user's interest in the web page. One limitation of collaborative filtering is cold start problem, i.e. we cannot predict interest (rating) for any web page. We can predict interest only for pages already visited by some users. The more users visit the web page the more precise the prediction is.

We compute the predicted interest of user u in an unvisited web page k as follows:

$$i_{u,k} = \bar{r_u} + \frac{\sum_{v=1}^{N} (r_{v,k} - \bar{r_v}) \times S_{u,v}}{\sum_{u=1}^{N} S_{u,v}}$$

where
$i_{u,k}$ is the predicted interest of user u in an unvisited web page k,
$\bar{r_u}$ is the average interest of user u in all web pages he visited,
$r_{v,k}$ is the interest of user v in the web page k,
$S_{u,v}$ is the similarity of users u and v,
N is the total number of users taken into account.

For measuring the similarity between two users we use Pearson correlation coefficient. It expresses the similarity as correlation between the ratings of two compared users. Therefore, we need at least one item which was rated by both users. In our case it means that both users should have visited at least one common web page. We compute the value of Pearson correlation coefficient as follows:

$$S_{u,v} = \frac{\sum_{i=1}^{I} (r_{u,i} - \bar{r_u}) \times (r_{v,i} - \bar{r_v})}{\sqrt{\sum_{i=1}^{I} (r_{u,i} - \bar{r_u})^2 \times \sum_{i=1}^{I} (r_{v,i} - \bar{r_v})^2}}$$

where
$S_{u,v}$ is the value of Pearson correlation coefficient of users u and v,
$\bar{r_u}$ is the average interest of user u in all visited web pages,
$r_{u,i}$ is the interest of user u in the web page i,
I are all pages visited by users u and v.

In our method we utilize the implicit feedback provided by the user when interacting with a web page. The feedback reflects the interest of the user, which we compute for visited web pages and predict for the other (unvisited) web pages. For this purpose we use collaborative filtering with new type of data. The advantage of this approach is that the user does not have to provide his interest explicitly, which removes the bias and inconsistencies among users' opinions. The interest is determined for every user using the same algorithm.

We strongly depend on the social context when determining the user's interest. The influence of two main interest indicators depends on the actions of other visitors of a web page. As we mentioned above, the limitation of this approach is that we are unable to determine the interest of the first visitor to a certain web page. We can do it using purely positive or negative interest indicators. The quality of interest estimation depends on the number of visitors to a certain web page. The more visitors there are, the more accurate the estimation will be.

The proposed method can generally be used across many environments. There are many interest indicators which can be tracked and used for interest computation. Our selection of time, number of scrolling actions and text copying events was influenced by the requirement, which determined that our solution should work in every web browser with no need for installation of add-ons. However, we may also use other interest indicators; in such case we need to set up their weights and their contribution to the total interest of the user, which is not entirely straightforward and requires some experimentation.

3.2 Discovering patterns in sequences of links

In each session, user visits several pages of the web portal. The session is described by a vector which elements are links to the web pages arranged in order they were visited. During visits to the web portal we create a long-term user model from these vectors. We assign weights to these vectors; vectors from earlier visits have lower weights.

Let session denote a continuous sequence of links during the user's visit to a web portal. In order to maintain the session's continuity, we use the referrer field of HTTP request message. If the URL of a previously visited page equals the referrer value of the currently visited page, we consider the pages to be in the same browsing session. Otherwise, we create a new session.

Similar users are detected based on the vectors of visited web pages. The process of dividing users into groups is presented in Algorithm 2. We consider

Algorithm 2 Group users according to their similarity

```
for all user $u$ do
  find patterns in clickstreams of $u$
  put $u$ to group according to prevailing pattern
end for
for all group $g$ do
  for all user $u$ in group $g$ do
    for all user $v$ in group $g$, $u \neq v$ do
      compute cosine similarity of clickstreams of users (u, v)
    end for
    sort users in group $g$ according to their similarity to $u$
  end for
end for
```

only the sequences of links within a website. Therefore, we need to repeat the grouping process for every website we want to personalize. It is not suitable to combine navigational patterns from different websites as they may have different structure of menus and navigational parts. However, it makes sense to detect different browsing strategies of users on the same website. Two users who are in the same group on portal A can be in different groups on portal B; the division of users into groups is valid only for particular website. This makes sense as the two portals can present totally different information and roles of the users on these portals may vary.

Groups of similar users are useful for recommendation of items among their members. We recommend links among similar users according to Algorithm 3.

Algorithm 3 Recommend links for user u from users similar to him

```
similar = select top $K$ similar users
for all user $v$ in similar  do
  calculate Pearson coefficient (u, v)
end for
for all page $p$ not visited by $u$ do
  predict interest of user in page ($u$, $p$)
end for
recommend top $M$ pages with highest predicted interest
```

Navigational patterns of users have to be of a certain minimal length (so that each sequence of two following pages does not represent a path pattern). We use four groups of users, each represented by one of the four different navigational patterns, and one group for users with no dominant pattern. After finding similar users to user u we select top K of them to form a recommendation group. The groups change according to new browsing sessions in which the users can behave differently. This reflects the possible evolution of user's behavior in time.

4 Evaluation

To evaluate the proposed method for user interest estimation we developed software tools which support adaptive navigation by recommending interesting web pages to guests of a web portal. We experimented with the web portal of our Faculty of Informatics and Information Technologies of the Slovak University of Technology in Bratislava (www.fiit.stuba.sk). It contains about 1,500 web pages with information for various types of visitors (students, teachers, alumni, and general public). The information needs of these users are diverse. However, the web portal of our faculty does not automatically differentiate among the users; each of them has to follow few links in order to get to “his” section.

We proposed a client-server architecture of the system with an adaptive proxy server [Barla and Bieliková 2010] in the middle, as shown in Figure 2. Adaptive proxy server developed at our faculty is an intermediary component through which users can access the web pages. As we can see from Figure 2, all requests from the users and responses from the target web servers go through the proxy server. Here, the requests and responses can be modified by various recommender systems, so that they are adapted to the user. This way we are able to personalize a website even though we do not have access to its source code and cannot alter it directly. The modifications of requests and responses are done via plug-ins which extend the basic functionality provided by the proxy server [Barla and Kramár 2011].

Adaptive proxy server provides user identification service, so that we can distinguish among different users. This is completely anonymous; the user is identified by a number with no further meaning, which is sent by the browser in every request as part of the user agent ID. This number is assigned upon setting up the web browser for using the proxy server. As a result, we are able to identify all web pages visited by a user, who is represented by a random number. We are unable to physically identify the user, unless he explicitly links his real identity with his numeric ID.

The proxy server can modify responses from web servers; we use it for embedding the behavior tracking script into every web page the users visit. This script sends collected data about user's behavior to the server. We track the amount of time actively spent on each web page, the number of scrolling events which occurred while the user was interacting with each web page, and the number of times the user copied some text into clipboard.

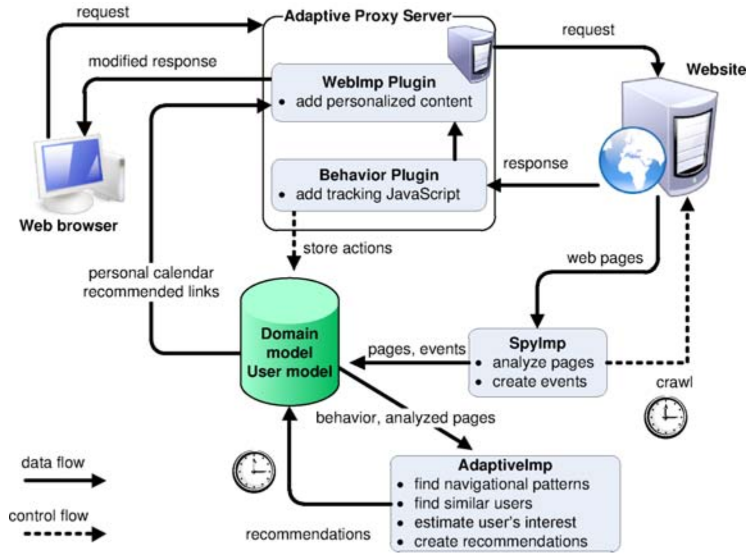


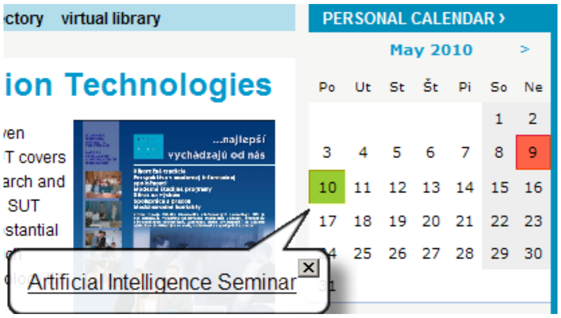
Figure 2: Architecture of our system for adaptation of navigation.

Data about behavior of users are analyzed by *AdaptiveImp*, which is a server component responsible for determination of the meaning of context-dependent interest indicators and computation of interest for each visited web page. Then, it predicts user's interest for a set of web pages which were visited by users similar to him. The similarity is computed from the sequences of links used, in which *AdaptiveImp* finds navigational patterns. The web pages with high value of predicted interest are selected as recommendations to the user. Recommendations are stored in a database in the way they should be displayed to a particular user. We run the jobs of *AdaptiveImp* twice a day. We do not need to compute the recommendations in real-time because a single user visits the web portal of our faculty few times a week. This may be different in other domains.

SpyImp is another server component, which creates the domain model by analyzing pages of the selected web portal. It crawls the web portal of our faculty once a day, since the new pages are not created very often. Each downloaded web page is parsed and its content is analyzed. In particular, we try to find dates on web pages and then extract the information about an upcoming event from it (as the web portal of our faculty contains mostly announcements about events). Each event is stored as an object with a hyperlink to the web page on which we found it. The interest associated with a user and this page is assigned to the event as well. When we predict the user's interest in an unvisited page, we associate it also with all events which can be found on this web page. Interesting events are then recommended in a personal calendar built for every user.

Some of the web pages change after they are published on the portal. The user might have seen the web page in the past and he is relying on the information he found on it. However, when this information changes, the user does not get any notification about it. He might easily overlook the change. Therefore, the web pages are monitored for changes using the *SpyImp* tool. We detect any change in the main content of the web page. This step might also involve more intelligent change detection, i.e. which change is just a correction of typographical error and which is an important change of the content. This is not within the scope of our research. If the value of user's interest in a certain web page is high and this page changes, it is put into the news section and recommended to the user for repeated visit.

The last component, *WebImp*, is a plug-in for the adaptive proxy server. Its purpose is to select sections with recommended items according to the user ID and add them to the web page. One of those sections is the calendar. It is dynamic and personalized to every user. It contains reminders (events which the user found interesting in the past) and recommendations (events which have high predicted value of interest). Figure 3 shows part of a web page with added personalized calendar. Another section is the news section which contains links to interesting pages (according to the user) which have changed.



Po	Ut	St	Št	Pi	So	Ne
					1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	29	30
31						

Figure 3: Screenshot of a personal calendar with recommended event highlighted.

We conducted a series of experiments on our faculty website in order to evaluate the proposed method of automatic interest estimation and subsequent link recommendation. We focused on these research questions:

- Does the automatic interest computation match real interest of the users?
- Can we find different navigational patterns dominating different users' browsing sessions?
- Do the users find recommended links interesting?

– Are the users satisfied with the personalized web page?

In the first experiment visitors to the faculty web portal were asked to explicitly state their interest in every visited web page as an integer from 0 to 10 (higher number means higher interest). They did know that their actions are being monitored, but they did not know which actions we recorded. This way we monitored the behavior of users on 55 web pages published on the portal. We automatically computed user's interest in each browsing session and compared it with provided rating.

The average accuracy of the interest computation was 62 %. Results indicate that time actively spent on a web page is the best interest indicator. This is analogous with the results of [Morita and Shinoda 1994]. Scrolling events proved to indicate positive interest as well. The experiment also showed that when a user does not scroll the web page it does not always mean he is not interested in it. Copying text into clipboard proved to be an entirely positive interest indicator. However, users did it only to a very small extent.

In order to evaluate the detection of navigational patterns we monitored the interaction of users with the website for longer period of time (5 weeks). We collected sequences of clicked links from every browsing session and selected those with the minimal length of 3 links. We got 52 browsing sessions of 19 different users. We analyzed each sequence of links in order to find navigational patterns in it. Then we put each user into a group according to the most dominant pattern in all his sessions. If there was no dominant pattern (e.g. the user's sessions contained equal amount of more than one type of pattern) we put the user into the fifth group. The numbers of users in each group are in Table 1.

Table 1: Splitting visitors into groups according to navigational patterns.

	path	**ring**	**loop**	**spike**	**no group**
number of users	7	1	3	3	5

We identified all four types of navigational patterns and found users for each of 5 groups. According to the patterns we can distinguish among users and the information task they are solving.

We also did an off-line experiment to prove the validity of the recommendations. We collected browsing activities from 24 users. We divided these data to testing and training sets (2 weeks each). For the purpose of this experiment all users were considered to be in the same recommendation group. We computed users' interests for every page in sessions from the training set. Then we predicted interest for pages each user has not visited. We selected top 10 pages with the highest predicted interest as recommended pages. Then we evaluated if the recommended pages were present in the testing set. We also estimated user's interest in each visited page that was previously recommended.

Across this experiment people visited 25 % of the pages recommended to them. In an analogous on-line experiment we recommended 38 links in total. Users explicitly expressed their interest in visiting 55 % of recommended links. Our goal during the link recommendation was to provide additional links to those the user normally uses. We did not perform a direct guidance of the user by telling him which link he should use next. In this context we consider the achieved results as success.

We also evaluated the overall satisfaction of the users with the personalized calendar using a questionnaire. The users stated that they liked the recommended events and that they would welcome other personalized sections as well. They also expressed their concerns about privacy when using the proxy server. The solution to this would be better explanation of the proxy server's functions as well as better availability of its source code so that users can verify its harmlessness.

5 Conclusions

User feedback is important element in personalizing the content of the Web. We focus on collecting implicit feedback and interest estimation based on it. Our method of automatic interest computation is based on tracking the actions the user takes during his browsing session on every web page he visits. We present how to use collaborative filtering with such data to predict potential interest of the user in unvisited web pages. The results we got indicate that this method can be utilized while recommending interesting links or web pages.

In this paper we show that for particular domains it is possible to estimate user's interest solely from his behavior while interacting with a certain web page. Our results can be further used in a mixed recommender system which also considers the content of the documents (i.e. textual content presented on each web page) [Kompan and Bieliková 2010], as well as the actions taken by the user while visiting that web page. Further extensions can include clustering based both on social network [Pham et al. 2011] or the content [Zelením and Bieliková 2011]. Such recommender may be employed by each portal which provides lots of content targeting various groups of users.

Our main contribution is a method for adaptive recommendation of interesting links. It is based on collaborative filtering in which we utilize recorded actions which we transform into the interest of a user. We are able to determine this interest automatically, without any explicit feedback. We also predict user's interest for yet unvisited pages and use it for link recommendation. The interest estimation can be used in various environments as a complement to the content-based recommender system.

The navigational patterns detection can be used for direct guidance of users which are new to the web portal. According to the sequence of links a user follows we can determine his actual intention (e.g. if he is exploring the web portal or if he is looking for one particular information).

In general, it is not possible to personalize any web portal as we do not have access rights to it. We also do not have means for studying the behavior of web users on the open Web directly. We proposed an elegant solution using a proxy server. Thanks to it we can add a behavior monitoring script to almost any website and thus record users' actions on it. Moreover, the proxy server allows us to modify the responses sent from the target web server to the client. We can also apply the modifications to a certain group of users.

We use our proxy server to personalize the web page by inserting sections with recommended links to it. Using a proxy server for accessing the web is a standard policy in business corporations so our method might be easily applied in this environment.

Acknowledgements

This work was supported by grants No. VG1/0508/09, VG1/0675/11, APVV 0208-10 and it is a partial result of the Research and Development Operational Program for the projects Support of Center of Excellence for Smart Technologies, Systems and Services, ITMS 26240120005 and Research and Development Operational Program for the projects Support of Center of Excellence for Smart Technologies, Systems and Services II, ITMS 26240120029, co-funded by ERDF.

The authors wish to thank all members of the adaptive proxy development team (members of Personalized Web group, pewe.fiit.stuba.sk) for their invaluable contribution to the proxy realization and deployment and helping with experiments during its use.

References

[Andrejko et al. 2006] Andrejko, A., Barla, M., Bieliková, M., Tvarožek, M.: "Tools for User Characteristics Acquisition"; Vojtáš, P. (ed.): Proc. of Annual Conf. Datakon'06 (2006), 139-148.

[Ahmed et al. 2009] Ahmed, S., Park, S., Jung, J.J., Kang, S.: "A Personalized URL Re-ranking Method using Psychological User Browsing Characteristics"; Journal of Universal Computer Science, 15, 4 (2009), 926-940.

[Baraglia et al. 2006] Baraglia, R., Lucchese, C., Orlando, S., Serrano, M., Silvestri, F.: "A Privacy Preserving Web Recommender System"; Proc. 2006 ACM Symposium on Applied Computing, ACM, New York (2006), 559-563.

[Baraglia and Silvestri 2007] Baraglia, R., Silvestri, F.: "Dynamic Personalization of Web Sites without User Intervention"; Comm. ACM, 50, 2 (2007), 63-67.

[Barla et al. 2009] Barla, M., Tvarožek, M., Bieliková, M.: "Rule-based User Characteristics Acquisition from Logs with Semantics for Personalized Web-based Systems"; Computing and Informatics, 28, 4 (2009), 399-427.

[Barla and Bieliková 2010] Barla, M., Bieliková, M.: "Ordinary Web Pages as a Source for Metadata Acquisition for Open Corpus User Modeling"; Proc. IADIS Int. Conf. on WWW/Internet, IADIS Press (2010), 227-233.

[Barla and Kramár 2011] Barla, M., Kramár, T.: "PeWeProxy Workshop."; Proc. of 9^{th} Workshop on Personalized Web – Science, Technologies and Engineering, STU Bratislava (2011), 75-86, http://pewe.fiit.stuba.sk/ontozur09-2011.pdf.

[Brusilovsky 1996] Brusilovsky, P.: "Methods and Techniques of Adaptive Hypermedia"; User Modeling and User-Adapted Interaction, 6, 2-3 (1996), 87-129.

[Cadez et al. 2003] Cadez, I., Heckerman, D., Meek, C., Smyth, P., White, S.: "Model-Based Clustering and Visualization of Navigation Patterns on a Web Site"; Data Mining and Knowledge Discovery, 7, 4 (2003), 399-424.

[Canter et al. 1985] Canter, D., Rivers, R., Storrs, G.: "Characterizing User Navigation through Complex Data Structures"; Behaviour & Information Technology, 4, 2 (1985), 93-102.

[Clark et al. 2006] Clark, L., Ting, I.H., Kimble, C., Wright, P., Kudenko, D.: "Combining Ethnographic and Clickstream Data to Identify User Web Browsing Strategies"; Information Research, 11, 2 (2006) http://informationr.net/ir/11-2/paper249.html.

[Claypool et al. 2001] Claypool, M., Le, P., Wased, M., Brown, D.: "Implicit Interest Indicators"; Proc. 6^{th} Int. Conf. on Intelligent User Interfaces, ACM, New York (2001), 33-40.

[Cockburn and McKenzie 2001] Cockburn, A., McKenzie, B.: "What Do Web Users Do? An Empirical Analysis of Web Use"; Int. Journal of Human-Computer Studies, 54, 6 (2001), 903-922.

[Dalamagas et al. 2007] Dalamagas, T., Bouros, P., Galanis, T., Eirinaki, M., Sellis, T.: "Mining User Navigation Patterns for Personalizing Topic Directories"; Proc. 9^{th} Annual ACM Int. Workshop on Web Information and Data Management, ACM, New York (2007), 81-88.

[Das et al. 2007] Das, A.S., Datar, M., Garg, A, Rajaram, S.: "Google News Personalization: Scalable Online Collaborative Filtering"; Proc. 16^{th} Int. Conf. on World Wide Web, ACM, New York (2007), 271-280.

[Fox et al. 2005] Fox, S., Karnawat, K., Mydland, M., Dumais, S., White, T.: "Evaluating Implicit Measures to Improve Web Search"; ACM Trans. on Information Systems, 23, 2 (2005), 147-168.

[Gauch et al. 2007] Gauch, S., Speretta, M., Chandramouli, A., Micarelli, A.: "User Profiles for Personalized Information Access"; Brusilovsky, P., Kobsa, A., Nejdl, W. (eds.): The Adaptive Web, Springer, Berlin (2007), 54-89.

[Goldberg et al. 1992] Goldberg, D., Nichols, D., Oki, B.M., Terry, D.: "Using Collaborative Filtering to Weave an Information Tapestry"; Comm. ACM, 35, 12 (1992), 61-70.

[Hofmann 2004] Hofmann, T.: "Latent Semantic Models for Collaborative Filtering"; ACM Trans. on Information Systems, 22, 1 (2004), 89-115.

[Hu et al. 2008] Hu, Y., Koren, Y., Volinsky, C.: "Collaborative Filtering for Implicit Feedback Datasets"'; Proc. 8^{th} IEEE Int. Conf. on Data Mining, IEEE Computer Society, Washington (2008), 263-272.

[Hu and Zhong 2005] Hu, J., Zhong, N.: "Clickstream Log Acquisition with Web Farming"; Proc. 2005 IEEE/WIC/ACM Int. Conf. on Web Intelligence, IEEE Computer Society, Washington (2005), 257-263.

[Jawaheer et al. 2010] Jawaheer, G., Szomszor, M., Kostkova, P.: "Characterisation of Explicit Feedback in an Online Music Recommendation Service"; Proc. 4^{th} ACM Conf. on Recommender Systems, ACM, New York (2010), 317-320.

[Kompan and Bieliková 2010] Kompan, M., Bieliková, M.: "Content-Based News Recommendation"; Lecture Notes in Business Information Processing, 61, Springer, Berlin (2010), 61-72.

[Krištofič and Bieliková 2005] Krištofič, A., Bieliková, M.: "Improving Adaptation in Web-based Educational Hypermedia by Means of Knowledge Discovery"; Proc. 16^{th} ACM Conf. on Hypertext and Hypermedia, ACM, New York (2005), 184-192.

[Lee and Brusilovsky 2009] Lee, D.H., Brusilovsky, P.: "Reinforcing Recommendation Using Implicit Negative Feedback"; Proc. 17^{th} Int. Conf. on User Modeling, Adaptation, and Personalization, Springer, Berlin (2009), 422-427.

[Li et al. 2010] Li, Y., Hu, J., Zhai, C., Chen, Y.: "Improving One-class Collaborative Filtering by Incorporating Rich User Information"; Proc. 19^{th} ACM Int. Conf. on Information and Knowledge Management, ACM, New York (2010), 959-968.

[Liu et al. 2010] Liu, N.N., Xiang, E., Zhao, M., Yang, Q.: "Unifying Explicit and Implicit Feedback for Collaborative Filtering"; Proc. 19^{th} ACM Int. Conf. on Information and Knowledge Management, ACM, New York (2010), 1445-1448.

[Manber et al. 2000] Manber, U., Patel, A., Robison, J.: "Experience with Personalization of Yahoo!"; Comm. ACM, 43, 8 (2000), 35-39.

[Mobasher 2007] Mobasher, B.: "Data Mining for Web Personalization"; Brusilovsky, P., Kobsa, A., Nejdl, W. (eds.): The Adaptive Web, Springer, Berlin (2007), 90-135.

[Morita and Shinoda 1994] Morita, M., Shinoda, Y.: "Information Filtering Based on User Behavior Analysis and Best Match Text Retrieval"; Proc. 17^{th} Annual Int. ACM SIGIR Conf. on Research and Development in Information Retrieval, Springer, New York (1994), 272-281.

[Niemann et al. 2010] Niemann, K., Scheffel, M., Friedrich, M., Kirschenmann, U., Schmitz, H.C., Wolpers, M.: "Usage-based Object Similarity"; Journal of Universal Computer Science, 16, 16 (2010), 2272-2290.

[Pham et al. 2011] Pham, M.C., Cao, Y., Klamma, R., Jarke, M.: "A Clustering Approach for Collaborative Filtering Recommendation Using Social Network Analysis"; Journal of Universal Computer Science, 17, 4 (2011), 583-604.

[Pan and Scholz 2009] Pan, R., Scholz, M.: "Mind the Gaps: Weighting the Unknown in Large-scale One-class Collaborative Filtering"; Proc.15^{th} ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining, ACM, New York (2009), 667-676.

[Pilászy and Tikk 2009] Pilászy, I., Tikk, D.: "Recommending New Movies: Even a Few Ratings are More Valuable than Metadata"; Proc. 3^{rd} ACM Conf. on Recommender Systems, ACM, New York (2009), 93-100.

[Rennie and Srebro 2005] Rennie, J.D.M., Srebro, N.: "Fast Maximum Margin Matrix Factorization for Collaborative Prediction"; Proc. 22^{nd} Int. Conf. on Machine Learning, ACM, New York (2005), 713-719.

[Resnick et al. 1994] Resnick, P., Iacovou, N., Suchak, M., Bergstrom, P., Riedl, J.: "GroupLens: An Open Architecture for Collaborative Filtering of Netnews"; Proc. 1994 ACM Conf. on Computer Supported Cooperative Work, ACM, New York (1994), 175-186.

[Sarwar et al. 2001] Sarwar, B., Karypis, G., Konstan, J., Reidl, J.: "Item-based Collaborative Filtering Recommendation Algorithms"; Proc. 10^{th} Int. Conf. on World Wide Web, ACM, New York (2001), 285-295.

[Sheng et al. 2008] Sheng, C., Hsu, W., Lee, M.L.: "Discovering Geographical-specific Interests from Web Click Data"; Proc. 1^{st} Int. Workshop on Location and the Web, ACM, New York (2008), 41-48.

[Suchal and Návrat 2010] Suchal, J., Návrat, P.: "Full Text Search Engine as Scalable k-nearest Neighbor Recommendation System"; IFIP WCC Series 331, Springer, Berlin (2010), 165-173.

[Sugiyama et al. 2004] Sugiyama, K., Hatano, K., Yoshikawa, M.: "Adaptive Web Search Based on User Profile Constructed Without any Effort from Users"; Proc. 13^{th} Int. Conf. on World Wide Web, ACM, New York (2004), 675-684.

[Tauscher and Greenberg 1997] Tauscher, L., Greenberg, S.: "How People Revisit Web Pages: Empirical Findings and Implications for the Design of History Systems"; Int. Journal of Human-Computer Studies, 47, 1 (1997), 97-137.

[Ting et al. 2005] Ting, I.H., Kimble, C., Kudenko, D.: "UBB Mining: Finding Unexpected Browsing Behaviour in Clickstream Data to Improve a Web Site's Design"; Proc. 2005 IEEE/WIC/ACM Int. Conf. on Web Intelligence, IEEE Computer Society, Washington (2005), 179-185.

[Velayathan and Yamada 2007] Velayathan, G., Yamada, S.: "Behavior Based Web Page Evaluation"; Proc. 16^{th} Int. Conf. on World Wide Web, ACM, New York (2007), 1317-1318.

[Xie and Phoha 2001] Xie, Y., Phoha, V.V.: "Web User Clustering from Access Log Using Belief Function"; Proc. 1^{st} Int. Conf. on Knowledge Capture, ACM, New York (2001), 202-208.

[Zehraoui et al. 2010] Zehraoui, F., Kanawati, R., Salotti, S.: "Hybrid Neural Network and Case Based Reasoning System for Web User Behavior Clustering and Classification"; Int. Journal of Hybrid Intelligent Systems, 7, 3 (2010), 171-186.

[Zeleník and Bieliková 2011] Zeleník, D., Bieliková, M.: "News Recommending Based on Text Similarity and User Behaviour"; Proc. of the 7^{th} Int. Conf. on Web Information Systems and Technologies, SciTePress (2011), 302-307.

[Zhou et al. 2008] Zhou, Y., Wilkinson, D., Schreiber, R., Pan, R.: "Large-Scale Parallel Collaborative Filtering for the Netflix Prize"; Proc. 4^{th} Int. Conf. on Algorithmic Aspects in Information and Management,Springer, Berlin (2008), 337-348.

[Zhu 2010] Zhu, T.: "Clustering Web Users Based on Browsing Behavior"; Proc. 6^{th} Int. Conf. on Active Media Technology, Springer, Berlin (2010), 530-537.