Usage-Based Positive and Negative Verification of User Interface Structure

Przemysław Kazienko
Wrocław University of Technology, Wyb.Wyspianskiego 27, 50-370 Wrocław, Poland
kazienko@pwr.wroc.pl, http://www.zsi.pwr.wroc.pl/~kazienko/eng_index.html

Abstract

The new approach to structure verification of the web hyperlinks is presented in the paper. It utilizes several types of patterns extracted from the web usage data: positive and negative association rules, positive sequential patterns as well as the new kind of patterns – sequential patterns with negative conclusions. All they enable to appraise the usefulness of hyperlinks in both the positive and negative way. Based on this knowledge, the content managers can adequately promote the trustworthy hyperlinks and remove the negatively verified ones.

1. Introduction

The structure of user interface in the web-based information systems is expressed by the set of hyperlinks used by users to move from one page to another. Hyperlinks are mostly created by content managers according to their knowledge and experience. On the other hand, users navigate though the site and leave fingerprints of their activities in the form of records in logs. We can make use of this usage data to verify usefulness of hyperlinks. It means that some patterns extracted from data about user behavior can be utilized to either acknowledge or deny the need of the existence of hyperlinks. This knowledge can be used by content managers to eliminate some ineffective hyperlinks or move the useful ones to the more prominent place in the web page.

2. Related Work

Web site navigational structure can be assessed upon several data sources: web usage, web content, knowledge of the web manager as well as the project fixed requirements and its general aims. This assessment may include many different factors. Almost every part of a web portal design may be assessed. There are many approaches to improve site structure usability; the basic one is link validation. Currently, every web design application has the ability to validate the hyperlink destination but the problem of missing links still occurs. Its innovative mobile agent solution, which can be used even with very limited access to the Internet connection, was proposed in [7].

The content-based navigational connections are derived from the similarity between textual content of web pages. The content in such approach is usually described with descriptors i.e. terms, which are expected to be the most informative and distinctive. One of the best known measures for descriptor selection is term frequency - inverse document frequency measure (*tf-idf*) [13, 21]. Terms, which occur relatively frequent in one document (*tf*), but rarely in the rest of the set (*idf*), are more likely to be relevant to the topic of the document. Terms that appear on many pages are not useful in distinguishing between a relevant page and irrelevant ones. The inverted document frequency *idf* reduces the influence of these terms. Moreover, terms with the too low or too high *idf*, as the bad content descriptors, can be excluded from further processing [13]. Besides, the importance of terms that occur in some specific parts of the HTML content like title, description and keywords can be increased [12]. A content-based system links and recommends pages similar to the just viewed one. It is kind of item-to-item correlation [18].

Baraglia and Silvestri utilized their own measure of web page usability. It is based on the analysis of web logs and is independent from the content of pages. The strength of the correlation between two pages is symmetric and it is, in its idea, similar to confidence function in association rules, see sec. 4.2. The main difference is that the authors used in the denominator the greater values from two: the number of user sessions containing the first web page and the number of sessions with the second page [5].

Proposals of new hyperlinks are the other aspect of navigation structure improvement. One of such methods exploits case based reasoning CBR as a possibility for the automatic generation of hyperlinks for hypertexts as extension of traditional textual methods [10].

The other method of hyperlink assessment is querying visitors using forms [17]. However, it is very difficult to evaluate the replies as users tend to present subjective opinions. Moreover, a site designer is not able to compare the results to a model site as one does not exist.

The next method is a statistical log analysis. It may deliver information about most common path, average ses-

sion length, pages where visitors leave the site, etc. An example of improvements based on this approach has been presented in [24].

There are also some mathematical methods to assess the navigation structure of an information system, e.g. the technique that utilizes the complexity of graph representing the site [27]. The most reliable graph measure appears to be a number of independent paths in the graph.

Srikant and Yang proposed algorithms to discover wrong locations of web pages in the hierarchical structure of the site based on the backtrack analysis of navigational paths extracted from web logs [23].

In many cases an automated assessment is the best way to discover incorrect site structure. Based on the archive of navigational patterns some automatic simplifications of path existing in the system can be recommended [20, 23]. Typical association rules and their indirect version were used for creation of recommendation ranking list and as the minor importance research for the assessment of hyperlinks. Conducted experiments revealed that about a half of all hyperlinks can be confirmed by typical association rules while almost 90% by indirect ones [14].

Spiliopoulou and Pohle have defined the success of a site as the efficiency of its component pages in attracting the users to exploit the supported services and buy the offered goods, especially in e-commerce sites. For this purpose, they proposed three basic measures: the contact efficiency, relative contact efficiency and conversion efficiency of a page. All of them are evaluated with statistical analysis of data about page requests and both customer and non-customer user sessions extracted from web logs. Finally, they recommended pages that needed to be improved [22].

Cowderoy analyzed the web site complexity from the developer perspective and compared it to the typical metrics useful for software development projects [9].

Effectiveness of the web site can be also studied from the organizational point of view as the measure for quality of services provided especially by the state dependent agencies [26].

The idea of both negative and positive verification of hyperlinks based on association rules derived from web logs has been proposed for the first time in [15] and then examined in [16]. Positive and negative patterns including simplified, 2-page sequential patterns extracted from usage data have been utilized for filtering of content-based recommendation list in [11].

3. Usage Data

Every http server is able to record and retain all incoming http requests in its log files. A typical request consists of information about the user (IP address, user agent, user identifier, and user password), the requested resource (URL) and the request itself (date and time, version of the protocol, http method, code, and size of the returned resource) as well as the navigation (referer field). At anonymous access user identifier and user password are left empty. Here we have the content of the example log:

1. 156.17.3.118 - - [14/Jan/2007:00:00:49 +0100] "GET /ebip/rss.php HTTP/1.0" 200 4228 "-" "MagpieRSS/0.72 (+http://magpierss.sf.net)"
2. 90.156.41.179 - - [14/Jan/2007:00:01:02 +0100] "GET /doktoranci/ HTTP/1.1" 200 3983 "http://www.pwr.wroc.pl/" "Mozilla/5.0 (Windows; U; Windows NT 5.1; en-US; rv:1.8.0.9) Gecko/20061206 Firefox/1.5.0.9"
3. 90.156.41.179 - - [14/Jan/2007:00:01:04 +0100] "GET /doktoranci/zapisy/ HTTP/1.1" 302 5 "http://www.pwr-old.wroc.pl/doktoranci/" "Mozilla/5.0 (Windows; U; Windows NT 5.1; en-US; rv:1.8.0.9) Gecko/20061206 Firefox/1.5.0.9"

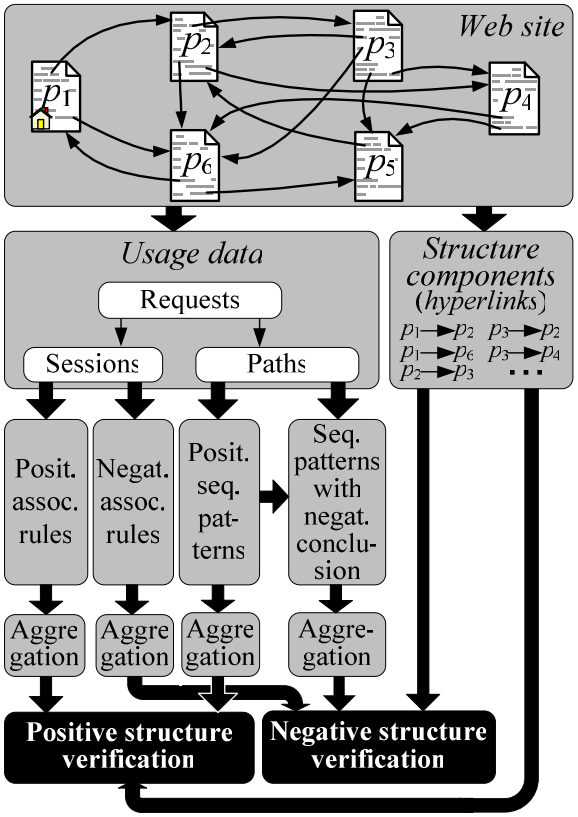

Figure 1. Positive and negative verification of user interface structure based on usage patterns

Most web sites in the Internet provide anonymous access to their pages. For that reason, sessions and paths, i.e. usage data, have to be extracted from the cleansed set of HTTP requests stored in the log files using some rules [8]. First, requests from web crawlers (robots), usually sent by search engines, have to be removed based either on textual analysis of the user agent field or on the entire set analysis [25]. Next, only meaningful resources are left, i.e. those from the white list of resource types: html, pdf, doc, etc.

Definition 1. Let $P=\{p_1, p_2, ..., p_N\}$ be a finite set of web pages in a single web site. The set s_i, called a session, is a tuple $s_i=<s_i^+,s_i^->$, where $s_i^+ \subset P$ is the set of distinct

pages visited during the ith users' visit in the web site and s_i^- is the compliment of s_i^+ in P, i.e. $s_i^-=P\backslash s_i^+$. Let S be a multiset of all available user sessions. Only sessions with $card(s_i^+)>1$ are taken into consideration.

Definition 2. The path $h_i=<p_{i1}, p_{i2}, \ldots, p_{im_i}>$ is the sequence of consecutive pages requested (visited) during the ith users' visit i.e. user session s_i; where m_i is the number of requests in session s_i. Only paths with $m_i>1$ are taken into consideration. Pages are ordered by increasing request time, i.e. page p_{ik+1} has been requested just after page p_{ik}, for all natural $k<m_i$. Let H be the multiset of all available paths.

In other words, a session is the partition of set P, i.e. $s_i^+\cup s_i^-=P$ and $s_i^+\cap s_i^-=\emptyset$. Since s_i^+ is the set of all distinct pages visited during the ith session and each page from s_i^+ may occur in the path h_i many times then $m_i\geq card(s_i^+)$.

Note that repetitions are allowed within both S and H, i.e. there may exist many sessions with the same content pages and many paths that would contain the same pages in the same order.

The simple but quite effective session and path extraction consists in matching IP address and user agent fields from the cleansed log data set, i.e. all request that came from the same IP address and user agent with the idle time between two following requests of less than 30 minutes ($t_{k+1}-t_k<30$min) [8, 6] are treated as one user session.

4. Verification of the Web-based User Interface Structure

4.1. The General Concept of Verification

The usage data – log files are processed to obtain sessions (unordered set of visited pages) and navigational paths, Fig. 1. Next, the positive and negative association rules are extracted from sessions. Simultaneously, paths are used to discover positive sequential patterns and the new kind of negative patterns – sequential patterns with negative conclusions. Since all the patterns operate on sets of items, they need to be aggregated to provide single values for pairs of pages. These aggregated values are used to verify hyperlinks extracted from the content of the web site in the either positive or negative way. Positive verification combines positive association rules and sequential patterns whereas negative association rules and sequential patterns with negative conclusions are used to discover useless hyperlinks that can later be removed by content managers.

4.2. Positive and Negative Association Rules Extracted from Usage Data

A positive association rule is an implication of the form $X\rightarrow Y$, where $X\subset P$, $Y\subset P$, $X\cap Y=0$. A rule $X\rightarrow Y$ means that if the page set X occurs in some user sessions then the page set Y co-occurs in these sessions. In other words, there are $N^+>0$ user sessions $s_i=<s_i^+,s_i^->$ in S for which $X\cup Y\subset s_i^+$.

Each rule has two associated measures that denote its significance and strength, called support and confidence respectively. The support $sup^a(X\rightarrow Y)$ of the positive rule $X\rightarrow Y$ in set S specifies the popularity of the rule among all sessions, i.e. in the entire set S. It is described with the following formula:

$$sup^a(X\rightarrow Y)=\frac{card\left(\left\{s_i=\left\langle s_i^+,s_i^-\right\rangle\in S: X\cup Y\subset s_i^+\right\}\right)}{card(S)}. \quad (1)$$

The confidence $conf^a(X\rightarrow Y)$ of a positive rule $X\rightarrow Y$ in set S is:

$$conf^a(X\rightarrow Y)=\frac{card\left(\left\{s_i=\left\langle s_i^+,s_i^-\right\rangle\in S: X\cup Y\subset s_i^+\right\}\right)}{card\left(\left\{s_i=\left\langle s_i^+,s_i^-\right\rangle\in S: X\subset s_i^+\right\}\right)}. \quad (2)$$

Another type of associations are negative rules [1, 3, 28, 29]. A confined negative association rule is a negative implication of the form $X\rightarrow\sim Y$, where $X\subset P$, $Y\subset P$, $X\cap Y=0$. A confined negative association indicates the negative relationship between X and Y in user sessions, i.e. there are $N^->0$ user sessions $s_i=<s_i^+,s_i^->$ in S that contain set X but do not contain set Y, i.e. $X\subset s_i^+$ and $Y\subset s_i^-$. The support $sup^a(X\rightarrow\sim Y)$ of a confined negative rule $X\rightarrow\sim Y$ in set S is:

$$sup^a(X\rightarrow\sim Y)=\frac{card\left(\left\{s_i=\left\langle s_i^+,s_i^-\right\rangle\in S: X\subset s_i^+\wedge Y\subset s_i^-\right\}\right)}{card(S)}. \quad (3)$$

The confidence $conf^a(X\rightarrow\sim Y)$ of a confined negative rule $X\rightarrow\sim Y$ in set S is:

$$conf^a(X\rightarrow\sim Y)=\frac{card\left(\left\{s_i=\left\langle s_i^+,s_i^-\right\rangle\in S: X\subset s_i^+\wedge Y\subset s_i^-\right\}\right)}{card\left(\left\{s_i=\left\langle s_i^+,s_i^-\right\rangle\in S: X\subset s_i^+\right\}\right)}. \quad (4)$$

As typically in association rules mining, only positive and negative rules with support and confidence that exceed two minimum thresholds $minsup^a$ and $minconf^a$, respectively, are considered.

There are several algorithms that extract both positive and negative association rules [1, 3, 28, 29] and some others designed for only positive rules extraction like apriori, Eclat, FP-growth.

4.3. Aggregation of Association Rules

In general, rules of the form $X\rightarrow Y$ or $X\rightarrow\sim Y$ operate on sets of elements, i.e. both X and Y can consist of many web pages. Moreover, there may be many rules $X\rightarrow Y$ for a pair of pages p_i and p_j that $p_i\in X_i$ and $p_j\in Y$. To utilize

association rules to assess pairs of user interface connections (hyperlinks) we would need a single measure for a pair of pages p_i and p_j. Thus, based on the confidence as usability measure, an integration mechanism has been introduced. It joins into one value average confidence $conf^{ass\text{-}avg}(p_i \rightarrow p_j)$ all positive association rules $X \rightarrow Y$ containing p_i on their left side ($p_i \in X$) and p_j on the right side ($p_j \in Y$), for every pair p_i and p_j, as follows:

$$conf^{ass\text{-}avg}(p_i \rightarrow p_j) = \frac{\sum_{X: p_i \in X, Y: p_j \in Y} conf^{a}(X \rightarrow Y)}{card(\{X \rightarrow Y : p_i \in X, p_j \in Y\})}. \quad (5)$$

Similarly, separately for every pair p_i and p_j, we consider all negative association rules $X \rightarrow \sim Y$ for which $p_i \in X$, $p_j \in Y$. Consequently, we obtain one average confidence value $conf^{ass\text{-}avg}(p_i \rightarrow \sim p_j)$:

$$conf^{ass\text{-}avg}(p_i \rightarrow \sim p_j) = \frac{\sum_{X: p_i \in X, Y: p_j \in Y} conf(X \rightarrow \sim Y)}{card(\{X \rightarrow \sim Y : p_i \in X, p_j \in Y\})}. \quad (6)$$

Note that for each pair p_i, p_j, only one of three mutually exclusive cases is possible:

1. There exists at least one positive association rule $X \rightarrow Y$, $p_i \in X$, $p_j \in Y$ which support and confidence exceed $minsup^a$ and $minconf^a$, respectively, and as a result $conf^{ass\text{-}avg}(p_i \rightarrow p_j) \geq minconf^a$ and the relation from p_i to p_j is positive.
2. There exists at least one negative association rule $X \rightarrow \sim Y$, $p_i \in X$, $p_j \in Y$ which support and confidence exceed $minsup^a$ and $minconf^a$, respectively, and as a result $conf^{ass\text{-}avg}(p_i \rightarrow \sim p_j) \geq minconf^a$ and the relation from p_i to p_j is negative.
3. There is neither positive $X \rightarrow Y$ nor negative association rule $X \rightarrow \sim Y$ such that $p_i \in X$, $p_j \in Y$ or these rules do not exceed minimum thresholds. Consequently, $conf^{ass\text{-}avg}(p_i \rightarrow p_j) = conf^{ass\text{-}avg}(p_i \rightarrow \sim p_j) = 0$, and we cannot say anything about relation from p_i to p_j.

4.4. Positive Sequential Patterns based on Usage Data

Sequential patterns are sequences of items (pages) frequently visited one after another by users within their navigational paths (see definition 2). To define frequent sequences we need to specify the concept of sequences inclusion.

Definition 3. A sequence $h_i = <p_{i1}, p_{i2}, \ldots, p_{im_i}>$ contains another sequence $q_j = <p_{j1}, p_{j2}, \ldots, p_{jn}>$, if there exist n integers $k_1 < k_2 < \ldots < k_n$, called index K of q_j in h_i, such that $p_{ik_1} = p_{j1}$, $p_{ik_2} = p_{j2}$, …, $p_{ik_n} = p_{jn}$. Sequence q_j is also called the subsequence of h_i. Item p_{ik_n} is called the end or the last item of q_j in h_i with respect to index K whereas its position in h_i, that means k_n, is called the end position and denoted by k^{end}.

Each sequence h_i may contain up to $(2^{m_i} - m_i - 1)$ different sequences q_j that consist of at least two items-pages. Note that shorter sequences (1-item) are completely useless in the structure verification.

Let us consider an example. A navigational path $h = <p_4, p_6, p_1, p_6>$ means that first the user visited page p_4, next page p_6, p_1, and they finished on page p_4. This sequence contains the following 2-, 3- and 4-item subsequences: $<p_4, p_6>$, $<p_4, p_1>$, $<p_4, p_6, p_1>$, $<p_4, p_1, p_6>$, $<p_4, p_6, p_6>$, $<p_4, p_6, p_1, p_6>$, $<p_6, p_1>$, $<p_6, p_6>$, $<p_6, p_1, p_6>$, $<p_1, p_6>$. Their number – 10 is less than the maximum (11) due to repetition of p_6. Note that subsequence $<p_4, p_6>$ have two separate indexes K_1 and K_2 for the given sequence h: $K_1 = (1,2)$ and $K_2 = (1,4)$; the end positions are $k_1^{end} = 2$ and $k_2^{end} = 4$, respectively. The index denotes positions of the subsequence's items within the given sequence.

Definition 4. Support $sup^{q+}(q)$ of sequence q in H (see definition 2) is the number of all source sequences (paths) from H that contain subsequence q. Support can be expressed either as an integer or the percentage of the H's quantity:

$$sup^{q+}(q) = \frac{card(\{h \in H : q \text{ is a subsequence of } h\})}{card(H)}. \quad (7)$$

If sequence q is contained frequently enough in the source sequences from H, i.e. $sup^{q+}(q) \geq minsup^{q+}$ then such sequence q is called a positive sequential pattern in H.

There are several algorithms to mine positive sequential patterns like AprioriSome, AprioriAll [2], PrefixSpan [19] or SPAM [4]. Any of them can be used to extract positive sequential patterns.

4.5. Sequential Patterns with the Negative Conclusions

Having positive sequential patterns specified, we can define an auxiliary compliment and finally a new pattern in data mining: sequential patterns with the negative conclusions.

Definition 5. A complement $C(q,h,K)$ of the subsequence q in sequence h with respect to index K is the set of all items of h that follows the last item of subsequence q in h. The most numerous complement of subsequence q in h is called the maximum complement of q in h and denoted with $C^{max}(q,h)$.

For the example navigational path $h = <p_4, p_6, p_1, p_6, p_1, p_6>$ and subsequence $q = <p_4, p_6>$, we have three end positions $k_1^{end} = 2$, $k_2^{end} = 4$, and $k_3^{end} = 6$ for three corresponding indexes K_1, K_2, and K_3, respectively. Hence, complement $C_1(q,h,K_1) = C_2(q,h,K_2) = \{p_1, p_6\}$ and $C_3(q,h,K_3) = \varnothing$. The complement is not a multiset so it does not allow repeti-

tions. The first two complements are the greatest so they are simultaneously the maximum complement $C_1(q,h,K_1)=C_2(q,h,K_2)=C^{max}(q,h)$. In general, the maximum complement corresponds to index K with the smallest value of end position $k_1^{end}=2$.

Note that for the given sequence h and its subsequence q, their maximum compliment contains all their complements: $C_1 \subseteq C^{max}$, $C_2 \subseteq C^{max}$, and $C_3 \subseteq C^{max}$. Based on this feature we can easily prove that if page p does not belong to maximum complement $C^{max}(q,h)$, then page p also does not belong to its subsets, i.e. to any of the complements $C_i(q,h,K_i)$.

Now, we can try to discover new patterns for each positive sequential pattern, namely sequential patterns with negative conclusion using all its maximum compliments.

Definition 6. A sequential pattern $s^-(q \rightarrow \sim X)$ with the negative conclusion in path multiset H is the implication $q \rightarrow \sim X$. It denotes that if sequence q occurs in the source paths h_i then set X does not intersect any of the complements in these paths h_i, i.e. there some paths $h_i \in H$: q is the subsequence of h_i and $X \cap C^{max}(q,h_i)=\emptyset$.

In other words, if the user visits sequence q of web pages they usually do not visit any of the pages from set X. It simultaneously means that set X occurs in maximum complement of sequence q in paths h from H very rarely. Term *rarely* denotes here that such case is valid for many source paths.

Similarly to association rules, each sequential pattern $s^-(q \rightarrow \sim X)$ with the negative conclusion possesses two measures: support $sup^{s-}(q \rightarrow \sim X)$ and confidence $conf^{s-}(q \rightarrow \sim X)$.

$$sup^{s-}(q \rightarrow \sim X) = \frac{card\left(\left\{h \in H : q \text{ is a subseq. of } h \wedge C^{\max}(q,h) \cap X = \emptyset\right\}\right)}{card(H)} \quad (8)$$

$$conf^{s-}(q \rightarrow \sim X) = \frac{card\left(\left\{h \in H : q \text{ is a subseq. of } h \wedge C^{\max}(q,h) \cap X = \emptyset\right\}\right)}{card\left(\left\{h \in H : q \text{ is a subsequence of } h\right\}\right)} \quad (9)$$

Note that $conf^{s-}(q \rightarrow \sim X) = sup^{s-}(q \rightarrow \sim X) / sup^{q+}(q)$.

Only patterns $s^-(q \rightarrow \sim X)$ that exceed minimum thresholds are really considered, i.e. $sup^{s-}(q \rightarrow \sim X) \geq minsup^{s-}$ and $conf^{s-}(q \rightarrow \sim X) \geq minconf^{s-}$.

A sequential pattern $sup^{s-}(q \rightarrow \sim X)$ with the negative conclusion, which has 1-page left side q, is simply equivalent to a negative association rule $q \rightarrow \sim X$, see sec. 4.2.

Sequential patterns with negative conclusion can be discovered from the previously obtained set of positive sequential patterns. First, the frequent set of maximum complement is extracted from source paths that contain each such positive sequential pattern. Pages from domain P (see definition 1) that do not belong to any complement of these paths automatically become members of the negative pattern conclusion X. All other pages that frequent maximum complement are treated as candidate members for conclusion set X. These candidates and their combinations X_i that exceed $minsup^{s-}$ and $minconf^{s-}$ thresholds form sequential patterns $s^-(q \rightarrow \sim X_i)$ with negative conclusions. The process is repeated for each positive sequential pattern q.

4.6. Aggregation of Sequential Patterns

Similarly to association rules, eq. (5) and (6), we can calculate average support $sup^{s+avg}(p_i,p_j)$ of positive sequential patterns, for each pair of web pages p_i and p_j. Separately, average confidence $conf^{s-avg}(p_i \rightarrow \sim p_j)$ is evaluated for sequential patterns with the negative conclusions.

4.7. Verification Functions

Aggregated confidences $conf^{ass-avg}(p_i \rightarrow p_j)$ of positive association rules and aggregated support of positive sequential patterns $sup^{s+avg}(p_i,p_j)$ are utilized to evaluate positive verification function $verif^+(p_i \rightarrow p_j)$ that corresponds to the connection from page p_i to p_j:

$$verif^+(p_i \rightarrow p_j) = \alpha * conf^{ass-avg}(p_i \rightarrow p_j) + \beta * sup^{s+avg}(p_i,p_j) \quad (10)$$

where α and β are constants that help to balance influence of association rules and sequential patterns.

Similarly, the negative verification function $verif^-(p_i \rightarrow p_j)$ is calculated based on average negative confidences of association rules $conf^{ass-avg}(p_i \rightarrow \sim p_j)$ and sequential patterns with negative conclusions $conf^{s-avg}(p_i \rightarrow \sim p_j)$, separately for each connection from p_i to p_j:

$$verif^-(p_i \rightarrow p_j) = \gamma * conf^{ass-avg}(p_i \rightarrow \sim p_j) + \delta * conf^{s-avg}(p_i \rightarrow \sim p_j). \quad (11)$$

where γ and δ are adjustment constants.

Based on the values of $verif^+(p_i \rightarrow p_j)$ and $verif^-(p_i \rightarrow p_j)$ we are able to verify usefulness of links between pages. The high value of $verif^+(p_i \rightarrow p_j)$ positively supports the existence of the hyperlink from p_i to p_j. Moreover, due to association rule contribution, the verification can even suggest new connections between pages.

On the other hand, a significant value of $verif^-(p_i \rightarrow p_j)$ can be an important sign for removal of the hyperlink from p_i to p_j. Since all components of verification functions are calculated from the usage data, i.e. http requests (navigational sessions and paths) then the entire verification process is based on historical user behaviors.

8. Conclusions and Future Work

Log files of the http servers contain data that can be used to discover patterns corresponding to behavior of web site users. Positive and negative association rules extracted from log data reflect typical usage patterns but they do not respect the navigational order whereas sequen-

tial patterns strongly depend on the sequence of navigation. Association rules and sequential patterns complement one another; by making use of the aggregated versions of them we obtain the comprehensive and compressed view onto how users utilize the structure of the web site, i.e. connections between pages.

The new pattern type – sequential patterns with negative conclusions provide new knowledge that enables to verify the existing hyperlinks in the negative way. As a result, the content manager of the web site can remove useless hyperlinks and simplify the structure of the user interface.

Future work will focus on the new algorithms that would help to mine both association rules and sequential patterns in the effective way.

Acknowledgment

This work was supported by The Polish Ministry of Science and Higher Education, grant no. N516 037 31/3708.

References

[1] Alataş B., Akin E.: An efficient genetic algorithm for automated mining of both positive and negative quantitative association rules. *Soft Computing - A Fusion of Foundations, Methodologies and Applications* 10(3) (2005) 230-237.

[2] Agrawal R., Srikant R.: Mining Sequential patterns. *11th ICDE*, IEEE Computer Society (1995) 3-14.

[3] Antonie M.-L., Zaïane O.R.: Mining Positive and Negative Association Rules: An Approach for Confined Rules. *PKDD 2004, LNCS 3202*, Springer Verlag (2004) 27-38.

[4] Ayres J., Gehrke J. E., Yiu T., Flannick J.: Sequential Pattern Mining Using Bitmaps. *Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining KDD 2002*, ACM, (2002) 429-435.

[5] Baraglia R., Silvestri F.: Dynamic personalization of web sites without user intervention. *Communication of the ACM* 50 (2) (2007) 63-67.

[6] Berendt B., Mobasher B., Spiliopoulou M., Wiltshire J.: Measuring the accuracy of sessionizers for web usage analysis. *Workshop on Web Mining at the First SIAM International Conference on Data Mining* (2001) 7-14.

[7] Chang W.-K., Chuang M.-H.: Validating Hyperlinks by the Mobile-Agent Approach. *Tunghai Science 3* (2001) 97-112.

[8] Chen Z., Fu A. W.-C., Tong F. C.-H.: Optimal Algorithms for Finding User Access Sessions from Very Large Web Logs. *World Wide Web: Internet and Web Information Systems* 6 (2003) 259-279.

[9] Cowderoy A. J. C.: Measures of size and complexity for web-site content. *11th ESCOM and 3rd SCOPE*, Munich, Germany (2000) 423-431.

[10] Haffner E. G., Heuer A., Roth U., Engel T., Meinel C.: Advanced Studies on Link Proposals and Knowledge Retrieval of Hypertexts with CBR. *EC-Web 2000, LNCS 1875*, Springer Verlag, (2000) 369-378.

[11] Kazienko P.: Filtering of Web Recommendation Lists Using Positive and Negative Usage Patterns. *KES2007, RAAWS 2007, LNAI 4694*, Springer Verlag (2007) 1016-1023.

[12] Kazienko P., Adamski M.: AdROSA - Adaptive Personalization of Web Advertising, *Information Sciences* 177 (11) (2007) 2269-2295.

[13] Kazienko P., Kiewra M., Personalized Recommendation of Web Pages. *Chapter 10* in: *Intelligent Technologies for Inconsistent Knowledge Processing*. Advanced Knowledge International, Adelaide, South Australia (2004) 163-183.

[14] Kazienko P., Kuźmińska K.: The Influence of Indirect Association Rules on Recommendation Ranking Lists. *ISDA 2005,* IEEE Computer Society (2005) 482-487.

[15] Kazienko P., Pilarczyk M.: Hyperlink Assessment Based on Web Usage Mining. *HT'06*, ACM Press (2006) 85-88.

[16] Kazienko P., Pilarczyk M.: Hyperlink Recommendation Based on Positive and Negative Association Rules. *New Generation Computing* 26 (3), May 2008, to appear.

[17] McGovern G., Norton R., O'Dowd, C.: *Web Content Style Guide*. Financial Times Press, Prentice Hall (2001).

[18] Mooney R. J., Roy L.: Content-based book recommending using learning for text categorization. *5th ACM Conference on Digital Libraries*, ACM Press (2000) 195-204.

[19] Pei J., Han J., Mortazavi-Asl B., Wang J., Pinto H., Chen Q., Dayal U., Hsu M.: Mining Sequential Patterns by Pattern-Growth: The PrefixSpan Approach. *IEEE Trans. on Knowledge and Data Eng.* 16 (11) (2004) 1424-1440.

[20] Rodriguez M. G.: Automatic data-gathering agents for remote navigability testing. *IEEE Software 19 (6)* (2002) 78-85.

[21] Salton G.: *Automatic Text Processing. The Transformation, Analysis, and Retrieval of Information by Computer*. Addison-Wesley, Reading, MA (1989).

[22] Spiliopoulou M., Pohle C.: Data Mining for Measuring and Improving the Success of Web Sites. *Data Mining and Knowledge Discovery, 5 (1/2)* (2001) 85-114.

[23] Srikant R., Yang Y.: Mining web logs to improve website organization. *WWW 10*, ACM Press (2001) 430-437.

[24] Sullivan T.: Reading Reader Reaction: A Proposal for Inferential Analysis of Web Server Log Files. *3rd Conf. on Human Factors and the Web*, US West Communications (1997).

[25] Tan P.-N., Kumar V.: Discovery of Web Robot Sessions Based on their Navigational Patterns. *Data Mining and Knowledge Discovery* 6 (2002) 9-35.

[26] Welch E.W., Pandey S.: Multiple Measures of Website Effectiveness and their Association with Service Quality in Health and Human Service Agencies. *40th Annual Hawaii International Conference on System Sciences HICSS*, IEEE Computer Society (2007) 107c.

[27] Weyuker E. J.: Evaluating software complexity measures. *IEEE Transactions on Software Engineering, 14 (9)* (1988) 1357-1365.

[28] Wu X., Zhang C., Zhang S.: Efficient Mining of Both Positive and Negative Association Rules. *ACM Transaction on Information Systems* 22 (3) (2004) 381-405.

[29] Yuan X., Buckles B.P., Yuan Z., Zhang J.: Mining Negative Association Rules. *ISCC'02*, IEEE Computer Society (2002) 623-628.