Multi-agent Web Recommendation Method Based on Indirect Association Rules

Przemysław Kazienko

Wrocław University of Technology, Department of Information Systems,
Wybrzeże S. Wyspiańskiego 27, 50-370 Wrocław, Poland
kazienko@pwr.wroc.pl, http://www.pwr.wroc.pl/~kazienko

Abstract. Recommendation systems often use association rules as main technique to discover useful links among the set of transactions, especially web usage data – historical user sessions. Presented in the paper new approach extends typical, direct association rules with indirect ones, which reflect associations existing "between" rather than "within" web user sessions. Both rule types are combined into complex rules which are used to obtain ranking lists needed for recommendation of pages in the web site. All recommendation tasks are distributed between many agents that communicate and transfer their knowledge one another.

1 Introduction

Recommendation systems become an important element of currently e-commerce and web based information systems [12]. Association rules implemented to user sessions or web server logs are one of the most popular data mining techniques used in such systems [11, 14] and they have been investigated in many papers [1, 2, 3, 10, 13, 15]. However, this method besides many advantages has also some restrictions that can lead to losing some vital information. Association rules denote relationships between web pages existing "within" common user navigational paths (sessions) but they omit relationships "between" sessions. It concerns associations between pages that rarely co-occur but there are many other common pages with which they appear together. This problem is especially important in the web environment where user requests usually result from selection of hyperlinks incorporated into the web page. In consequence, some close related pages, which are not connected with links, have fewer opportunities to be visited in common sessions.

Proposed in this paper method of recommendation includes indirect association rules that extend typical, direct ones. A multi-agent architecture, in which expert-agents may be distributed between many hosts and cooperate with one another, was implemented to make the system scalable, flexible and easy to extend [4]. Each agent is responsible for a single recommendation task, so it encapsulates specific functions that would be available for the rest of the system. Agents not only interchange information, but they also possess their own knowledge. The presented method is closely related to the web recommendation ROSA project [5, 6, 7, 8].

2 Multi-agent Architecture, Method Overview

Every system's expert-agent possesses its own characteristic depending on its role in the recommendation process (Fig. 1).

User Session Monitor captures users' HTTP requests and groups them into sessions using JSP servlet session mechanism [6]. It preserves data about active user session and sends it (the set of pages visited during the session) to Session Pre-processor just after the session has finished.

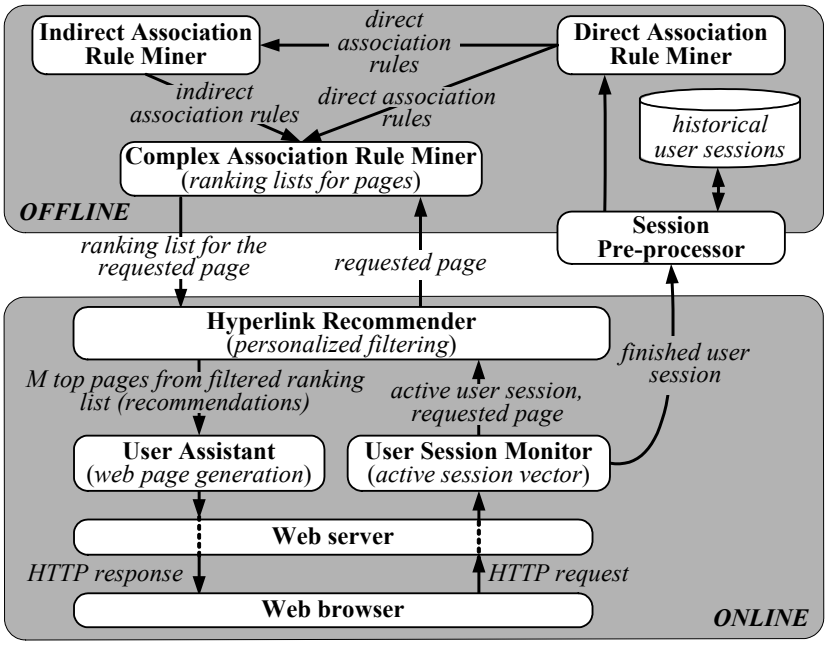

Fig. 1. Multi-agent architecture

Session Pre-processor filters and gathers in its own database finished sessions obtained form User Session Monitor - it omits too short sessions e.g. containing less than two HTTP requests. Storing and filtering is performed online. However, Session Pre-processor makes historical user sessions accessible for offline association rules mining. Thus, this agent works both online and offline.

The main recommendation process is performed offline and involves four agents (Fig. 2): Session Pre-processor, Direct Association Rule Miner, Indirect Association Rule Miner, and Complex Association Rule Miner. The only task for Session Pre-processor is to deliver historical user sessions to Direct Association Rule Miner.

Direct Association Rule Miner extracts useful association rules from user sessions using well-known mining algorithms: apriori [2] or incremental ones - FUP [3] or DLG [10, 15]. Such typical rules are called *direct association rules*. The mining process is restricted to only simplest rules which associate single web pages with other single pages. Appropriate parameters: minimal support and minimal confidence are used for filtering direct rules that seem to be useful.

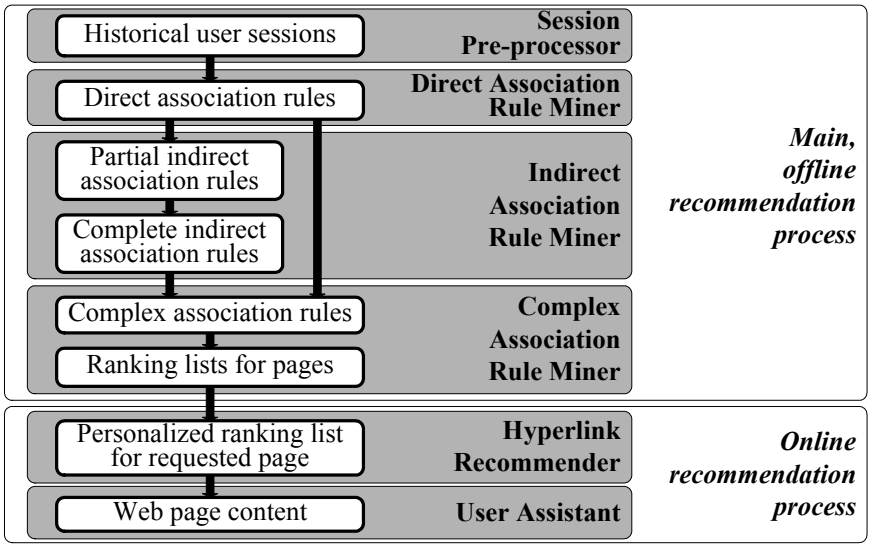

Fig. 2. Recommendation process based on mining of association rules

Indirect Association Rule Miner receives direct association rules from Direct Association Rule Miner and using IDRAM algorithm [9] calculates indirect association rules. Direct rules result from frequent co-occurrence of two pages in the same user sessions. However, two pages that relative frequent occur in common sessions with the third, transitive page are also supposed to be associated. Such associations are extracted as indirect rules and they are estimated using direct association rules rather than source user session data. To discover a *partial* indirect association rule, leading from one page (body) to another one (head) through the third page (transitive), two direct rules provided by Direct Association Rule Miner are used: the first one from the body to transitive page and the second one from the transitive to head page. Since it may exist many transitive pages for one pair body→head (many partial indirect rules), Indirect Association Rule Miner should take into consideration all of them. In this way, we obtain one *complete* indirect association rule for a pair of web pages. Similarly to direct association rules, complete indirect association rules are filtered using separate minimum confidence threshold.

Complex Association Rule Miner combines into *complex association rules* both direct and indirect rules delivered by Direct and Indirect Association Rule Miner, respectively. It creates a ranking list for each web page: head pages of complex rules with the greatest confidence value are on the top of ranking list for body page of the rule. Complex Association Rule Miner evaluates offline and stores separate ranking list for each page in the web site.

Hyperlink Recommender is responsible for creating hyperlink ranking list for the page currently requested by the active user. It receives the active user session data and the requested page (URL) from User Session Monitor. The requested page is relayed to Complex Association Rule Miner in order to obtain static ranking list for this page based on complex association rules. Next, the ranking list is filtered by Hyperlink Recommender to exclude pages lately visited by the user, using active user session

data. M top pages from the filtered ranking list are presented to the user (by means of User Assistant).

User Assistant generates the final web page content for the active user. The HTML content includes M hyperlinks (recommendations) provided by Hyperlink Recommender.

Since web usage tend to change over time, offline obtained association rules should be periodically recalculated. The knowledge update problem was solved by introduction of special update method into multi-agent architecture [6].

3 Association Rules

Definition 1. Let d_i be an independent *web page* (document) and D be the web site content (web page domain) that consists of independent web pages $d_i \in D$.

Definition 2. A set X of pages $d_i \in D$ is called a *pageset X*. A pageset does not contain repetitions: $\forall(d_i,d_j \in D)\ d_i,d_j \in X \Rightarrow d_i \neq d_j$.

Definition 3. A user session $S_i \in S^S$ is the *pageset* S_i containing pages viewed by the user during one visit in the web site; $S_i \subseteq D$; S^S – the set of all user sessions gathered by the system. Each session must consist of at least two pages $card(S_i) \geq 2$. A session contains the pageset X if and only if $X \subseteq S_i$.

Sessions correspond to transactions in typical data mining approach [2, 13].

Definition 4. A *direct association rule* is the implication $X \rightarrow Y$, where $X \subseteq D$, $Y \subseteq D$ and $X \cap Y = \emptyset$. Direct association rule is described by two measures: support and confidence. The direct association rule $X \rightarrow Y$ has the support $sup(X \rightarrow Y) = sup(X \cup Y)/card(S^S)$; where $sup(X \cup Y)$ is the number of sessions S_i containing both X and Y; $X \cup Y \in S_i$. The confidence *con* for direct association rule $X \rightarrow Y$ is the probability that the session S_i containing X also contains Y: $con(X \rightarrow Y) = sup(X \cup Y)/sup(X)$; $sup(X)$ – the number of sessions that contain the pageset X.

Direct association rules represent regularities discovered from a large data-set [1]. The problem of mining association rules is to extract all rules that have support and confidence greater than given thresholds: minimum direct support *supmin* and minimum direct confidence *conmin*.

Dependencies only between pagesets with the cardinality of 1 - single web pages are considered in this paper. For that reason the pageset X including d_i ($X = \{d_i\}$) will be represented by d_i.

An indirect association rule reflects the relationship between two pages coming out of the existence of another, "third" pages (transitive pages) with which these two pages separately have common sessions (Fig. 3). These indirect associated two pages do not need to occur together in any user session like in direct rules. Indirect association rule is called *partial*, if it concerns only one transitive page or *complete*, if it encapsulates all transitive pages.

Definition 5. *Partial indirect association rule* $d_i \rightarrow^{P\#} d_j,d_k$ is the *indirect* implication from d_i to d_j with respect to d_k, for which exist two direct association rules: $d_i \rightarrow d_k$ and $d_k \rightarrow d_j$ with $sup(d_i \rightarrow d_k) \geq supmin$, $con(d_i \rightarrow d_k) \geq conmin$ and $sup(d_k \rightarrow d_j) \geq supmin$, $con(d_k \rightarrow d_j) \geq conmin$, where $d_i,d_j,d_k \in D$; $d_i \neq d_j \neq d_k$. The page d_k, in the partial indirect

association rule $d_i \rightarrow^{P\#} d_j, d_k$, is called *the transitive page*. Each indirect association rule is described by *partial indirect confidence* $con^{P\#}(d_i \rightarrow^{P\#} d_j, d_k)$, as follows:

$$con^{P\#}(d_i \rightarrow^{P\#} d_j, d_k) = con(d_i \rightarrow d_k) * con(d_k \rightarrow d_j) \quad (1)$$

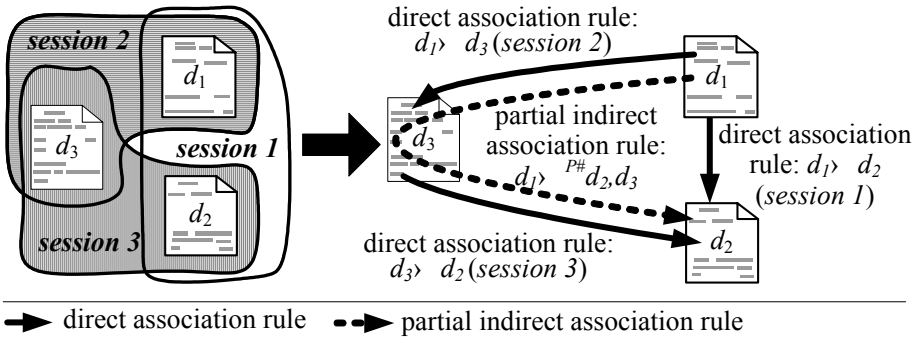

Fig. 3. Partial indirect association from the web pages d_1 to the page d_2

Definition 6. The set of all possible transitive pages d_k for which partial indirect association rule from d_i to d_j exists, is called T_{ij}.

Definition 7. *Complete indirect association rule* $d_i \rightarrow^{\#} d_j$ aggregates all partial indirect association rules from d_i to d_j with respect to all existing transitive pages $d_k \in T_{ij}$ and it is characterized by *complete indirect confidence* - $con^{\#}(d_i \rightarrow^{\#} d_j)$:

$$con^{\#}(d_i \rightarrow^{\#} d_j) = \frac{\sum_{k=1}^{card(T_{ij})} con^{P\#}(d_i \rightarrow^{\#} d_j, d_k)}{\max_T} \quad (2)$$

where $\max_T = \max_{d_i, d_j \in D} \left(card(T_{ij})\right)$.

All direct and indirect rules that exceed appropriate thresholds (support and confidence) are combined into one coherent complex association rules set.

Definition 8. *Complex association rule* $d_i \rightarrow^{*} d_j$ from d_i to d_j exists, if direct $d_i \rightarrow d_j$ or complete indirect $d_i \rightarrow^{\#} d_j$ association rule from d_i to d_j exists. Complex association rule is characterized by *complex confidence* - $con^{*}(d_i \rightarrow^{*} d_j)$, as follows:

$$con^{*}(d_i \rightarrow^{*} d_j) = \alpha * con(d_i \rightarrow d_j) + (1-\alpha) * con^{\#}(d_i \rightarrow^{\#} d_j) \quad (3)$$

where: α - direct confidence reinforcing factor, $\alpha \in [0,1]$. Setting α we can emphasize or damp the direct confidence at the expense of the complete indirect one.

4 Recommendation Process

Session Pre-processor collects historical user sessions and delivers them to Direct Association Rule Miner, which discovers direct association rules $d_i \rightarrow d_j$, using one of well-known algorithms [2, 3, 13, 10] (Fig. 2). Only rules that exceed *supmin* and *conmin* are accepted. Indirect Association Rule Miner uses IDARM algorithm [9] to

extract indirect rules and calculates complete indirect confidence for all existing indirect associations based on direct confidence (1) and aggregating all partial indirect rules (2). Only those indirect rules that have complete indirect confidence greater than given method parameter *indirconmin* ≥ *conmin*[2] are provided for further processing.

The complex confidence is estimated by Complex Association Rule Miner for each web page pair $d_i, d_j \in D$. If for such a pair exists a complex association rule, combined from rules delivered by Direct and Indirect Association Rule Miner, than (3) is used. Otherwise, $con^*(d_i \rightarrow^* d_j)=0$. In this way, we obtain the static, separate ranking list of pages – candidates for recommendation, with appropriate measure (complex confidence), for each web page. Since the offline process is periodically repeated because of new user sessions, ranking lists may evolve.

To personalize received lists the active user session should be monitored by User Session Monitor that stores not only the whole sequence of current user requests but also pages recommended on each page. To limit the amount of necessary data, only K pages lately visited by the user are kept in the extended form i.e. with the list of recommendation for each page. In this way, we retain the separate M x K matrix of URLs for each active user, where M is the maximum number of recommended pages.

Let L_k be the set (list) of pages recommended by the system on the k-th page of the active user, k=1, 2, …, K. The last visited page, from which the user came to the just being generated page, has the index 1, previous - 2, etc. This information maintained by User Session Monitor is also accessible for Hyperlink Recommender.

The ranking function $con^*(d_i \rightarrow^* d_j)$ for the requested page d_i and separately for every recommendation candidate d_j is recalculated by Hyperlink Recommender using *personalized ranking function* $r(d_i \rightarrow^* d_j)$ in order to damp pages that belong to L_k:

$$r\left(d_i \rightarrow^* d_j\right) = \begin{cases} con\left(d_i \rightarrow^* d_j\right) * \prod_{k: d_j \in L_k} \left(\frac{k-1}{K}\right), & \text{if } \exists k : d_j \in L_k \\ con\left(d_i \rightarrow^* d_j\right), & \text{otherwise} \end{cases} \quad (4)$$

The ranking function is here reduced with the factor (k-1)/K. For previous page (viewed just before the recently requested) this factor is equal to 0 (k=1) and web pages, that have been suggested on such a page, are excluded from current recommendation. Note that a page d_j may be recommended on many pages within the last K ones. For such a page d_j its ranking function value is decreased several times by (k-1)/K factor, although it does not prevent such a page to be recommended. The recommendation is possible while its ranking function value $con^*(d_i \rightarrow^* d_j)$ is much greater than for other pages. We are only sure that the now recommended page will not be suggested on the next page visited by the user (k will than be equal to 1).

5 Conclusions and Future Work

The proposed method exploits information about historical user behaviour to recommend local pages in the web site. Discovered direct and indirect association rules coming out of web user sessions are used to estimate helpfulness of an individual page for recommendation on the just visited web page. Obtained ranking lists are filtered and adapted to the current user behaviour, with respect to visited

pages. Particular recommendation tasks are performed by specialized agents that communicate one another.

The future work will concentrate on the adaptation of the method to e-commerce product recommendation and the integration with web content mining [7, 8].

References

1. Agrawal R., Imieliński T., Swami A.: Mining association rules between sets of items in large databases. Proceedings of the 1993 ACM SIGMOD International Conference on Management of Data, ACM Press (1993) 207-216.
2. Agrawal R, Srikant R.: Fast Algorithms for Mining Association Rules. Proceedings of the 20th International Conference on Very Large Databases (1994) 487–499.
3. Cheung D.W.L., Lee S.D., Kao B.: A General Incremental Technique for Maintaining Discovered Association Rules. Proceedings of the Fifth International Conference on Database Systems for Advanced Applications (DASFAA). Advanced Database Research and Development Series 6 World Scientific (1997) 185–194.
4. Ferber J.: Multi-Agent Systems: An Introduction to Distributed Artificial Intelligence. Addison Wesley Longman, England, (1999).
5. Kazienko P., Kiewra M.: Link Recommendation Method Based on Web Content and Usage Mining. New Trends in Intelligent Information Processing and Web Mining Proceedings of the International IIS: IIPWM´03 Conference, Advances in Soft Computing, Springer Verlag (2003) 529-534. http://www.zsi.pwr.wroc.pl/~kazienko/pub/IIS03/pkmk.pdf
6. Kazienko P., Kiewra M.: ROSA - Multi-agent System for Web Services Personalization. AWIC 2003. First International Atlantic Web Intelligence Conference Proceedings, LNAI 2663, Springer Verlag (2003) 297-306.
7. Kazienko P., Kiewra M.: Personalized Recommendation of Web Pages. Chapter in: Nguyen T. (ed.) Modern Technologies of Artificial Intelligence for Information Processing. Advanced Knowledge International, Australia (2004) to appear.
8. Kazienko P., Kiewra M.: Integration of Relational Databases and Web Site Content for Product and Page Recommendation. 8th International Database Engineering & Applications Symposium. IDEAS '04, IEEE Computer Society (2004).
9. Kazienko P., Matrejek M.: Indirect Association Rules in Web Usage Mining. (2004) to appear.
10. Lee G., Lee K.L., Chen A.L.P.: Efficient Graph-Based Algorithms for Discovering and Maintaining Association Rules in Large Databases. Knowledge and Information Systems, Vol. 3, No. 3, Springer Verlag (2001) 338-355.
11. Mobasher B., Cooley R., Srivastava J.: Automatic Personalization Based on Web Usage Mining. Communications of the ACM, Volume 43, Issue 8 (2000) 142-151.
12. Montaner M., López B., de la Rosa J. L.: A Taxonomy of Recommender Agents on the Internet. Artificial Intelligence Review, Vol. 19, Issue 4 (2003) 285-330.
13. Morzy T., Zakrzewicz M.: Data mining. Chapter 11 in Błażewicz J., Kubiak W., Morzy T., Rubinkiewicz M (eds): Handbook on Data Management in Information Systems. Springer Verlag, Berlin Heidelberg New York (2003).
14. Yang H., Parthasarathy S.: On the Use of Constrained Associations for Web Log Mining. WEBKDD 2002 - MiningWeb Data for Discovering Usage Patterns and Profiles, LNCS 2703, Springer Verlag (2003) 100 – 118.
15. Yen S.J., Chen A.L.P.: An Efficient Approach to Discovering Knowledge from Large Databases. Proceedings of the Fourth International Conference on Parallel and Distributed Information Systems. IEEE Computer Society (1996) 8–18.