Web Page Recommendation Based on Semantic Web Usage Mining

Soheila Abrishami[1], Mahmoud Naghibzadeh[2], and Mehrdad Jalali[1]

[1] Department of Computer Engineering, Azad University of Mashhad, Mashhad, Iran
{soheilaabrishami,jalali}@mshdiau.ac.ir
[2] Department of Computer Engineering, Ferdowsi University of Mashhad, Mashhad, Iran
naghibzadeh@um.ac.ir

Abstract. The growth of the web has created a big challenge for directing the user to the Web pages in their areas of interest. Meanwhile, web usage mining plays an important role in finding these areas of interest based on user's previous actions. The extracted patterns in web usage mining are useful in various applications such as recommendation. Classical web usage mining does not take semantic knowledge and content into pattern generations. Recent researches show that ontology, as background knowledge, can improve pattern's quality. This work aims to design a hybrid recommendation system based on integrating semantic information with Web usage mining and page clustering based on semantic similarity. Since the Web pages are seen as ontology individuals, frequent navigational patterns are in the form of ontology instances instead of Web page addresses, and page clustering is done using semantic similarity. The result is used for generating web page recommendations to users. The recommender engine presented in this paper which is based on semantic patterns and page clustering, creates a list of appropriate recommendations. The results of the implementation of this hybrid recommendation system indicate that integrating semantic information and page access sequence into the patterns yields more accurate recommendations.

Keywords: Web usage mining, semantic web, ontology, web page recommendation, page clustering.

1 Introduction

In semantic web content and information is interpretable and understandable not only by humans, but also by computers. In order to support the user in his task, the Web should be enriched by the machines' ability to process the information [1]. Therefore, the Web content and objects should be semantically introduced into the machine world by using ontologies. Ontologies have been created to facilitate knowledge reuse and sharing in the decentralized and distributed context of the Web [5].

The rapidly growing amount of information on the Web causes difficulties for the Internet users to find desired information and due to the huge volume of data and lack of structure in many Web sites finding relevant information on the Web has become a

K. Aberer et al. (Eds.): SocInfo 2012, LNCS 7710, pp. 393–405, 2012.

real challenge. Therefore, the research field of Web usage mining has gained notable consideration for finding user behavioral patterns. Web usage mining is concerned with finding user navigational patterns on the World Wide Web by extracting knowledge from web logs. One of the most important disadvantages of the current approaches in Web usage mining is that the result is produced in terms of Web pages (i.e. web page addresses); hence, there is no semantic meaning of the common navigation profile. Another disadvantage of classical web usage mining is called the new-item problem, which is the failure to recommend newly added pages or products to the visitors since these products or pages are not in the current common navigation profiles. To overcome this, the common navigation profile can be extracted in terms of semantic information so that the common navigation profile will be in ontological terms and concepts. Therefore, newly added items can be recommended to the user as long as this item's concept is in the common navigation profiles.

From that point, Web site content has started to play a more important role in the Web usage mining process, relying only on Web usage data for user modeling can be inefficient. In recent years, there have been studies where semantic knowledge systems are used in order to provide further improvement in accuracy.

The extracted patterns in web usage mining are useful in several different areas, including recommendation, Web site restructuring, prefetching, etc. Recommendation is a heavily studied research subject, where web usage patterns can improve the accuracy of the task.

In this work, we present a hybrid web recommender system, which incorporates semantic knowledge and sequence information into pattern generation and clusters Web pages by using a constructed ontology for Web site. The information of clustering is used for identifying of irrelevant pages in recommendation set. In [3] a hybrid recommender system is proposed, which integrates document clustering and semantic knowledge with web usage mining. In another hybrid system [4] user sessions are clustered and the navigational patterns are generated without using semantic information and then semantic features, which extracted from domain specific ontologies combined with the navigational patterns. However, in our work, semantic information is used within pattern generation, rather than in postprocessing, and we cluster the web pages based on semantic similarity among individuals in ontology which is created for the Web site. The results of the implementation of our hybrid recommender system show a significant improvement on the quality of recommendation engine output.

The rest of the paper is organized as follows. In Section 2, related work is presented. In Section 3, the details of the proposed approach are described. Section 4 includes the experimental work. Finally, Section 5 presents the concluding remarks.

2 Related Work

In the area of web usage mining, there are various approaches such as clustering, association rule mining, sequence mining, and sequence alignment to mine the data in the server logs. However, those works which combine web usage mining and semantic web are limited. Bettina Berendt, Andreas Hotho and Gerd Stumme [1, 17] are the authors of one of the first studies of web usage mining on the Semantic Web. In [16] they show how the Semantic Web can improve web usage mining, and how usage mining can help to build up the Semantic Web. In their work, they assumed that the server log contains terms of ontology concepts and individuals, so the mining algorithms like clustering and association rule mining can be applied on it, but there are no details about generation of pattern by using semantic information.

Mobasher et al., [9] have provided a system for incorporating domain ontologies with Web usage mining and in this work the emphasis is on the personalization process and the frequent pattern generation is done by clustering.

The work presented by Adda et al., [14] proposes using metadata about the content that they assume is stored in domain ontology to enhance the quality of the discovered patterns. Authors use two-level taxonomy, while the first taxonomy branches between concepts, the second one branch between relations among concepts, and this increases the time complexity of the algorithm considerably.

Recent studies presented in [13, 18] aim to enhance association rules with domain ontologies. In [18] each web server log file entry converts into a single ontology concept. After applying SPADE algorithm on the converted log file, sequential association rules are generated. Unlike standard sequential rule miners, this approach yields rules that have ontological objects as antecedent and consequent and the result is used for generating web page recommendations to the visitor. On the other hand, in the work presented in [13], the emphasis is on improving the time efficiency of online next page prediction, rather than pattern quality. In [19] they integrate semantic web into web usage mining and clustering of sessions is used for extracting navigational patterns.

3 Proposed Approach

The general architecture of the system has the basics of classical web usage mining systems. The initial steps include the data acquisition and data cleaning. These steps are followed by offline mining part and lastly, the recommendation phase which is the online part of the system.

As illustrated in Fig. 1, the proposed system consists of 4 phases: preprocessing, rule extraction, page clustering, and recommendation. In the rest of this section, the steps of the process are described in more detail.

3.1 Preprocessing

For Web Log Preprocessing, the entries of the Web server log is cleaned, transactions are extracted, and ontology class individuals are mapped to the Web page addresses. The preprocessing method which is used in [2] is chosen for the preprocessing phase our system.

The first phase of the log file parsing is the pruning of the non-responded web requests and eliminating requests made by software agents such as Web crawlers.

After pruning, the navigation history of each session from the log file is extracted. For extraction of navigation history User Identification and Session Identification must be done.

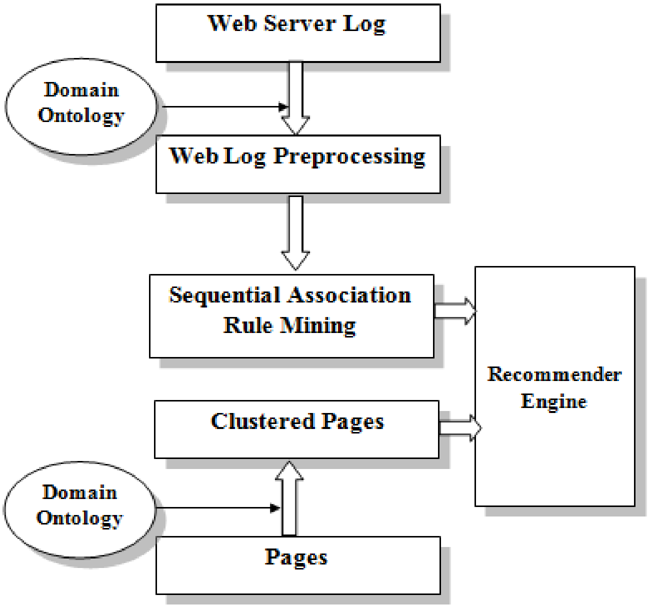

Fig. 1. General Framework of Overall System Design

We identify users with their IP address registered in each record in log file. For Session Identification, we should define session timeout. We assumed 20 minutes session timeout for the experimental procedure.

In order to prevent any postprocessing, in the last step, each page access in every transaction is mapped to the ontology instances defined in the ontology for the web site. Since the automatic extraction of semantic information is a hard task, and it is a research problem on its own, we accomplished this task manually. At first, we construct ontology in OWL associated with our web site by helping Web page structures. The construction of this basic ontology can be achieved through knowledge extraction from Web sites, based on content and structure mining. For mapping the suitable concepts and instances is chosen by considering the content of the Web pages.

Construction and creation of ontology are done using Protégé [11]. This ontology contains the concept of InfoResource, Science and FreeResource in first level. The Science concept includes a hierarchy relation in which higher lever categories include more details than lower level categories (for example Agriculture concept has a child named Forestry). After construction of ontology, the visited Web pages' URLs in navigation history are mapped to one or more individuals of the ontology. This mapping is done according to the semantic annotation of the web pages and web objects on the pages. At the end of this step, transactions consist of ontology individuals instead of pages' URLs.

3.2 Navigational Patterns Creation

After the preprocessing step, the next step is the extraction of frequent navigation patterns. Association rule mining[1] is used to discover interesting relations among items in a large database. Sequential association rule mining[2] is a limited version of association rule mining. In sequential association rule mining, items in both input (the dataset) and output (association rules) are presented in a sequential form (mostly ordered by a timestamp). In this research, we preferred to use sequential association rule mining since the sequence information in the navigation is retained in the generated patterns. Through sequential rule mining, the results will be in a sequential form that permits us to know which pages are visited and in what order. Since taxonomies are encountered in many ontology definitions and classes may have $is - a$ relations, it is necessary to find generalized sequence patterns under a given taxonomy. In order to find generalized association rules, the dataset is extended in such a way that each item is also associated with the item's taxonomical parent. For efficient and scalable mining of sequential patterns, PrefixSpan [7] is used. The PrefixSpan's input includes transactions which each transaction is a sequence of events. A sequence is an ordered set of events, and event is a set of one or more items. In Fig. 2, a sequence database is shown.

Sequence_id	***Sequence***
10	*<a(abc)(ac)d(cf)>*
20	*<(ad)c(bc)(ae)>*
30	*<(ef)(ab)(df)cb>*
40	*<eg(af)abc>*

Fig. 2. A sequence database [7]

SPMF [12] is an open-source data mining framework written in Java. For extracting sequential association rule mining, we used RuleGen [6] algorithm in this framework. So the input of this algorithm is an extended dataset, such taxonomies are introduced into the dataset, and each transaction is a sequence of events.

Some examples for the generated frequent sequence rules are presented in Fig. 3. The rule NaturalResource => Irrigation, LiveStock, NaturalResource tells that visiting concept "NaturalResource" will be followed by visits to the concepts of Irrigation, LiveStock and NaturalResource.

---

[1] The support of rule X ⇒Y is the percentage of the transactions that contain both X and Y to all transactions. The confidence is the percentage of the transaction that contains Y to the transaction that contains X.

[2] For a sequential association rule, a → b ⇒ c, a → b is the head of the rule and c is the body of the rule. When body is concatenated to the head, we end up with the sequence a → b → c. Therefore, support for this rule is ((number sessions including a → b → c) / (all transactions)) and confidence for this rule is ((number of sessions including a → b → c) / (number of sessions including a → b)).

Sample Rules
NaturalResource => *Irrigation , LiveStock , NaturalResource*
Gardening => *Gardening , Irrigation , LiveStock , NaturalResource*
LiveStock , AgricultureEngineering => *AgricultureEngineering , Foresty , Irrigation , LiveStock , NaturalResource*
Irrigation => *AgricultureEngineering , Foresty , Irrigation*

Fig. 3. Sample rules for the library Web site

3.3 Page Clustering By Using Semantic Similarity

In this step, we clustered URLs which were visited by users. We used K-means [20] clustering algorithm for grouping the pages. As described before, in this work, we mapped pages into a set of ontological item sets, so we clustered pages by using semantic similarities between ontology's individuals. For example, we have pages P_1 and P_2 which are mapped into certain individuals.

$P1.html$ is mapped into $obj2, obj3, obj4$

$P2.html$ is mapped into $ob\,j8, ob\,j15, ob\,j17$

Where each P_i is a unique page and each Obj_k is ontology individual.

In [19], an approach, which clusters user sessions that have been mapped into sequence of objects, is presented. In our research, we employed and modified this proposed approach to cluster the pages and used the result of this clustering in order to increase accuracy of our recommendation. In our method, although clustering is used, instead of sessions, pages are clustered. Since our clustering is based on semantic similarity between ontology's individuals, the result of page clustering reflects the similarity of pages. The outcome of clustering is used for identification of irrelevant pages in recommendation phase.

In order to cluster these pages we need two instruments:

- A distance metric to compute the distance between two ontology concepts.
- A distance metric to compute the distance between two ontology instances.

Since instances which are used in mapping are string types, we use "LevenshteinDistance" string comparison algorithm.

The distance between two concepts can be defined as a function of the distance between their attributes and their location in the ontological tree. We assume a concept to be a tuple $C = (A, L)$, where A is set of attributes $(a1, a2, a3, \dots, an)$ and L is a location representation in the taxonomy of the ontology. The distance between two concepts $C1$ and $C2$, $DIST(C1, C2)$, can be defined as the weighted sum of distances of object attributes and tree locations.

$$Dist(C1, C2) = DistA(att1, att2) * w1 + DistL(loc1, loc2) * w2 \quad (1)$$

where $DistA$ is a function that returns the distance between two sets of attributes and $DistL$ is a function that returns the distance between two locations in a tree; $w1$ and $w2$ are weights of the distance functions. For $DistL$ we use the approach by [8] that is based on the positions of the concepts in the concept taxonomy. For $DistA$ function, we employ the string distance definition "LevenshteinDistance".

By using these metrics for computing distances in the K-means algorithm, the pages are clustered. At the end of this phase, we have clustered URLs according to their semantic information.

3.4 Recommender Engine

This is the final component of the system. It combines the analysis of the usage mining and Web page clustering and produces recommendations for current users. The current user's navigation path is compared to all sequential association rules to produce recommended pages. For each recommended page, the page is checked to determine in which cluster it is located. After defining maximum clusters, which are clusters with a larger number of pages, the pages of these clusters are added to the final recommendation set.

Since the association rules are composed of ontology individuals, the user navigation history is converted into the sequence of ontology instances. In the recommendation phase, firstly, according to the window_count (window_count is a parameter which defines the maximum number of previously visited pages which should be used in order to recommend a new page) navigated items are taken as the search pattern. The association rules and user navigation history are joined and the consequent part association rules, whose antecedent part is equal to the search pattern, are extracted and added to the recommendation set. The ontology instances in the rule consequents are mapped back into Web page addresses. As a result, the number of pages in recommendation set increases. In previous work [18], the number of recommended pages is large and it is possible that some irrelevant pages are recommended as well.

In this research, by using information related to page clustering we identify irrelevant pages and only the pages which are unrelated will be deleted. Before these pages are recommended to a user with respect to page clustering information and how many of these pages are located in each cluster, the maximum clusters are selected and the pages which belong to these clusters are recommended to the user. This process is expressed in Algorithm 1. As an example, consider the active user navigation $Web\,page{:}IEEE \rightarrow Web\,page{:}ACM$ where these two pages are concrete Web pages with URLs. In the first step, user navigation history is mapped into ontology instances. As the result of this step, user navigation is turned to $ElectronicEngineering \rightarrow ComputerEngineering$. Assume that the only rule we have is $EngineeringMath \Rightarrow CivilEngineering$. Since $EngineeringMath$ is the parent of $ComputerEngineering$, it can be used for recommendation. At last, $CivilEngineering$ is mapped back into pages. If we assume these pages are p3, p5, p7, p9, p2, p4 and we have a page clustering like this:

$$C1{:}\,p1, p6, p8, p4 \qquad C2{:}\,p2, p10, p5 \qquad C3{:}\,p9, p3, p7$$

Pages p3, p7, p9 are located in C3, p2, p5 are located in C2 and p4 is located in C1. Since the number of clusters is three and based on the proposed algorithm, two clusters which are maximum, are selected and their pages are recommended to the user. Since two clusters C3 and C2 are maximum, Web pages p9, p3, p7, p2, p5 are added to the final recommendation set. By doing so, the number of recommended pages is reduced according to semantic similarity.

Algorithm 1. construct recommendation set(R, T, K, window_count, Clustered Pages)

Comment: Constructing recommendation set algorithm
Comment: R is set of association rules T is active user navigation path K is number of cluster

$for\ each\ rule\ R = < a_1, a_2, \dots a_n > \Rightarrow < c_1, c_2, \dots c_j >$
{
If $(a_n = t_m\ and\ a_{n-1} = t_{m-1}\ and \dots a_{n-window\_count} = t_{m-window\_count})$
then L = L ∪ R;
}
If$(K > 2)$
{ $for\ each\ page\ in\ L = p_1, p_2, \dots p_l$
Identify cluster p_i
Sort the clusters based the number of pages exist in them
If $(3 \leq K \leq 7)$
then select 2 clusters are maximum and add pages of them to final recommendation set
If $(K == 9)$
then select 3 clusters are maximum and add pages of them to final recommendation set
If $(K == 11)$
then select 4 clusters are maximum and add pages of them to final recommendation set
}

4 Evaluation Measures and Experimental Result

4.1 Evaluation Measures

Precision and coverage [15] are the two popular metrics that are used to evaluate and compare the performance of the proposed system.

Each testing session (ts) of testing set is divided into two parts. The first n web pages of session ts is used as the input of the recommendation engine which is denote as Recomment_List and the second part is simulated as the future requests (page visits) which are compared with the output of the recommendation system and denote as Real_List.

The precision of a transaction is given as the number of web pages correctly predicted divided by the total number of web pages predicted.

$$Precision_t = \frac{|Recomment\_List_{\ t} \cap Real\_List_{\ t}|}{|Recomment\_list_{\ t}|} \quad (2)$$

The coverage of a transaction is given as the number of web pages correctly predicted divided by the total number of web pages visited by the user.

$$Coverage_t = \frac{|Recomment\_List_{\ t} \cap Real\_List_{\ t}|}{|Real\_List_{\ t}|} \quad (3)$$

The precision and coverage were evaluated for all the transactions in the testing dataset and their averages were calculated. The average precision and average coverage values helped to evaluate the system.

In order to get a single evaluation measure, the M-metric, define in [10] is used.

$$M = \frac{2*coverage*precision}{coverage+precision} \tag{4}$$

4.2 Experimental Results

In this work, the experiments were conducted on the navigation logs that belong to library Web site of Ferdowsi University of Mashhad (http://c-library.um.ac.ir). We selected that Web pages related to Information Resources and Open Access parts. After the preprocessing, the log file includes 1300 sessions and contains 5,200 Web page views. The average number of Web pages in a session is 4. As described before, the ontology model of these parts consists of three concepts InfoResource, Science and FreeResource in first level which Science concept has a taxonomy levels.

The dataset is divided into two parts. The first part is the training part, 75% of the dataset, and the other part is reserved for testing, included the remaining 25% of the dataset. By applying the association rule mining algorithm on training data set, association rules are generated. In addition, by using K-means algorithm on mapped pages, clusters are determined. In our experiments, we used K=9 for K-means clustering since the best results are obtained by this value for K.

The first set of experiments shows that using semantic information improves the rule and recommendation quality. As it is seen in the table 1, in part (a) the resulting precision, coverage and matching rate values are higher than the result shown in part (b). As it was expected, using semantic information improves pattern and recommendation quality. In part (b) since semantic information does not contribute to the pattern structure in this experiment, the generated patterns and recommendations have low precision and coverage values.

Table 1. Comparison of the proposed system with recommender system without semantic web, window_count=1, min_ confidence=1, k=9

	Support	*Precision*	*Coverage*	*M*
Results by Applying Semantic and page clustering K=9 (a)	0.43023	0.496032	0.161199	0.243323
	0.33720	0.603175	0.214802	0.316789
	0.23255	0.603175	0.214802	0.316789
	0.19767	0.603175	0.214802	0.316789
	0.13953	0.515873	0.169488	0.255148
Results without Semantic (Just URL) (b)	0.0697	0.0238095	0.0010131	0.00194363
	0.0581	0.08333	0.028255	0.0422019
	0.046511	0.0809524	0.0285255	0.0418899
	0.03488	0.166667	0.037723	0.0608728
	0.02325	0.153886	0.0734597	0.0994469

In second set of experiments, we compared our work with approach presented in [18] by applying on our log file. In the first experiments, we have evaluated the effect of change in minimum support on precision values by using window_count=1 and comparison between our approach and method [18]. In our experiment and as Fig. 4 shows initially, precision value increases when the minimum support increases. This is due to the facts that, when the minimum support increases, weaker association rules are eliminated, and more accurate recommendations can be generated. However, after a breaking point, precision starts to decrease, since the number of association rules and recommendations decrease. Fig. 4 illustrates better result for precision recommendation in our system over [18]. As it was mentioned before, in [18] the number of pages, which are recommended, is large. However, in our method by using page clustering unrelated pages in recommendation sets are identified and those pages which are unrelated are omitted. Consequently, a more precise recommendation set can be presented to users.

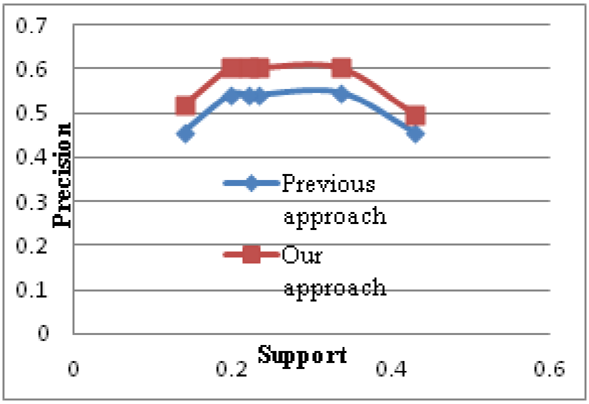

Fig. 4. Precision comparison of the proposed system with the system in [18], window_count=1, min_ confidence=1, k=9

Fig. 5 shows coverage comparison between our work and the work in [18]. As Fig. 5 depicts, the coverage of our system is less than the previous system. The reason for this is that in the system [18] when an individual is mapped back to URL's page, the number of pages which are obtained is large, the coverage metric is increased accordingly. However, in our system, using page clustering, we eliminate irrelevant pages and in some parts the number of recommended pages is reduced noticeably consequently, coverage metric is reduced as well. This reducing of extra pages exerts positive effect, and as it's obvious in Fig. 4, the precision metric is also improved.

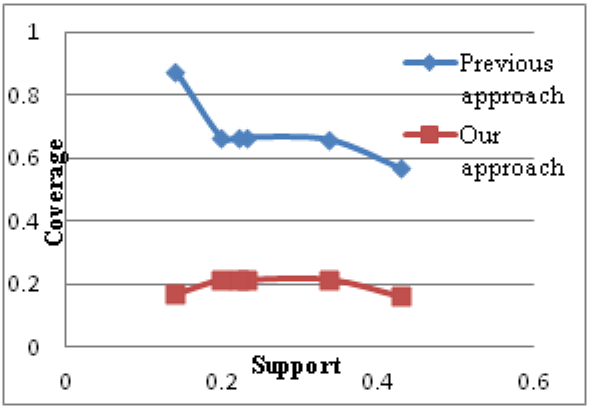

Fig. 5. Coverage comparison of the proposed system with the system in [18], window_count=1, min_ confidence=1, k=9

In the last group of experiments, the recommendations are generated under window_count=2 to display the effect of window count in results. Fig. 6 demonstrates that the precision asset value in both method is less than precision values in window_count=1, but as we expected our method for precision metric works better than [18] in window_count=2 as well.

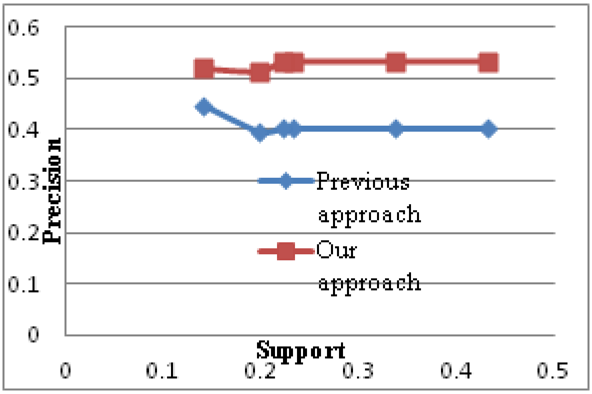

Fig. 6. Precision comparison of the proposed system with system in [18], window_count=2, min_ confidence=1, k=9

Fig. 7, shows coverage values as same as precision are decreased too and our approach has lower values than the [18] like window_count=1. In last set of experiments important observation is that the increase in window count has an insignificant and negative effect on the precision and the coverage, hence most recent visit appears to be the most effective one on the recommendation.

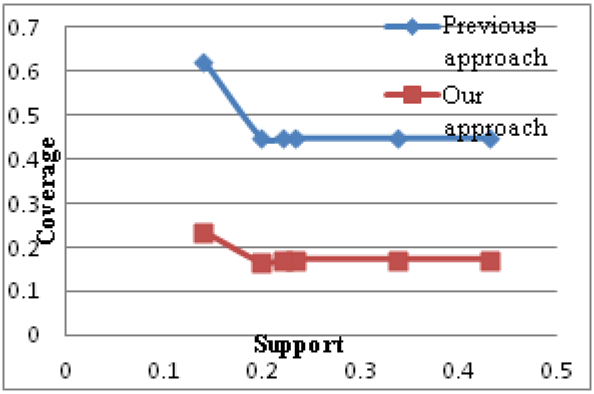

Fig. 7. Coverage comparison of the proposed system with system in [18], window_count=2, min_ confidence=1, k=9

5 Conclusion

Extracted pattern in Web usage mining has an important role in many areas, including recommendation, Web personalization, Web construction, Website organization and Web user profiling. In this paper, we have proposed an approach to extract user navigation behavior by using semantic web usage mining. To achieve this aim, we incorporate semantic web into generated patterns. With this combination the created rules contain ontology individuals instead of web pages' URLs.

The success of the system is measured by evaluation of the recommendations. In the presented work, improved pattern quality is used to recommend pages more accurately. Since the number of recommended pages is large, unrelated pages must be determined and omitted. Information from page clustering based on semantic similarity is used for identifying irrelevant pages which are obtained by rules and added to the recommendation set. In previous works, the number of pages which were recommended was large but in this research by using page clustering, that pages which are unrelated will be omitted so, the number of pages is narrowed down. The reducing number of pages affect positively. Therefore, the precision of our proposed system is enhanced. In this way, recommender engine, by using the created rules, selects some pages to recommend but before that irrelevant pages are eliminated according to page clustering. Our experiments suggest that integrating semantic knowledge with web usage mining and in addition to clustering of pages by using of concept similarity in ontology, can indeed be useful in recommender systems.

References

1. Berendt, B., Hotho, A., Stumme, G.: Towards Semantic Web Mining. In: Horrocks, I., Hendler, J. (eds.) ISWC 2002. LNCS, vol. 2342, pp. 264–278. Springer, Heidelberg (2002)
2. Cooley, R., Mobasher, B., Srivastava, J.: Data preparation for mining world wide web browsing patterns. Knowledge and information systems 1, 5–32 (1999)

3. Wei, L., Lei, S.: Integrated Recommender Systems Based on Ontology and Usage Mining. Active Media Technology, 114–125 (2009)
4. Samizadeh, R., Ghelichkhani, B.: Use of semantic similarity and web usage mining to alleviate the drawbacks of user-based collaborative filtering recommender systems use. International Journal of Industrial Engineering and Production Research (IJIE), English (2010)
5. Etminani, K., Delui, A.R., Naghibzadeh, M.: Overlapped ontology partitioning based on semantic similarity measures. In: 2010 5th International Symposium on Telecommunications (IST), pp. 1013–1018. IEEE (2010)
6. Zaki, M.J.: SPADE: An efficient algorithm for mining frequent sequences. Machine Learning 42, 31–60 (2001)
7. Pei, J., Han, J., Mortazavi-Asl, B., Wang, J., Pinto, H., Chen, Q., Dayal, U., Hsu, M.C.: Mining sequential patterns by pattern-growth: The prefixspan approach. IEEE Transactions on Knowledge and Data Engineering 16, 1424–1440 (2004)
8. Maedche, A., Zacharias, V.: Clustering Ontology-Based Metadata in the Semantic Web. In: Elomaa, T., Mannila, H., Toivonen, H. (eds.) PKDD 2002. LNCS (LNAI), vol. 2431, pp. 383–408. Springer, Heidelberg (2002)
9. Dai, H., Mobasher, B.: Integrating semantic knowledge with web usage mining for personalization. WebMining: Applications and Techniques.[20082 06211] (2009), `http://maya.cs.depaul.edu/~mobasher/papers/DM042WM2Book.pdf`
10. Mobasher, B., Dai, H., Luo, T., Nakagawa, M.: Improving the effectiveness of collaborative filtering on anonymous web usage data. In: IJCAI 2001 Workshop on Intelligent Techniques for Web Personalization ITWP01 2001 (2001)
11. `http://protege.stanford.edu/`
12. `http://www.philippe-fournier-viger.com/spmf/index.php`
13. Mabroukeh, N.R., Ezeife, C.I.: Using domain ontology for semantic web usage mining and next page prediction. In: Information and Knowledge Management, pp. 1677–1680. ACM (2009)
14. Adda, M., Valtchev, P., Missaoui, R., Djeraba, C.: Toward recommendation based on ontology-powered web-usage mining. IEEE Internet Computing 45–52 (2007)
15. Nakagawa, M., Mobasher, B.: Impact of site characteristics on recommendation models based on association rules and sequential patterns. In: IJCAI 2003 Workshop on Intelligent Techniques for Web Personalization (2003)
16. Stumme, G., Hotho, A., Berendt, B.: Usage Mining for and on the Semantic Web: next generation data mining. In: NSF Workshop (2002)
17. Stumme, G., Hotho, A., Berendt, B.: Semantic web mining: State of the art and future directions. Web Semantics: Science, Services and Agents on the World Wide Web 4, 124–143 (2006)
18. Senkul, P., Salin, S.: Improving pattern quality in web usage mining by using semantic information. Knowledge and information systems 30, 527–541 (2012)
19. Yilmaz, H., Senkul, P.: Using Ontology and Sequence Information for Extracting Behavior Patterns from Web Navigation Logs. In: 2010 IEEE International Conference on Data Mining Workshops, pp. 549-556. IEEE (2010)
20. MacQueen, J.: Some methods for classification and analysis of multivariate observations. In: 5-th Berkeley Symposium on Mathematical Statistics and Probability, California, USA, p. 14 (1967)