© 2005 - 2009 JATIT. All rights reserved.

EFFICIENT WEB USAGE MINING BASED ON FORMAL CONCEPT ANALYSIS

D. VASUMATHI [1] & Dr. A GOVARDHAN [2]

[1] Associate Professor, Department of Computer Science and Engineering, JNTU College of Engineering. Kukatpally, Hyderabad-500 085, Andhra Pradesh, India.

[2] Principal, JNTU College of Engineering. Jagityala, Karimnagar(Dist), Andhra Pradesh, India.

E-mail: vasukumar_devara@yahoo.co.in[1]

ABSTRACT

Web usage mining attempts to discover useful knowledge from the secondary data obtained from the interactions of the users with the web. Web usage mining has become very critical for effective web site management, creating adaptive web sites, business and support services, personalization and so on. Web usage mining aims to discover interesting user access patterns from web logs. Formal Based Concept Analysis(FBCA) is an effective data analysis technique based on ordered lattice theory. Formal Based Concept can then be generated and interpreted from the concept lattice using FBCA.FBCA has been applied to a wide range of domains including conceptual clustering, information retrieval and knowledge discovery.

In this paper, we propose a novel FBCA approach for web usage mining. In our approach, the FBCA technique is applied to mine association rules from web usage lattice constructed from web logs. The discovered knowledge(association rules) can then be used for practical web applications such as web recommendation and personalization. We apply the FBCA-mined association rules to web recommendation and compare its performance with that of classical Apriority -mined rules. The results indicate that the proposed FBCA approach not only generates far fewer rules than Apriority-based algorithms, the generated rules are also of comparable quality with respect to three objective performance measures.

Key words: *Web intelligence, Knowledge discovery, Data mining, Association Rules, Formal based concept Analysis, Web usage mining and Knowledge-based Systems.*

1. INTRODUCTION

The WWW continues to grow at an amazing rate as an information gateway and as a medium for conducting business. Web mining is the extraction of interesting and useful knowledge and implicit information from artifacts or activity related to the WWW. Based on several research studies we can broadly classify web mining into three domains: content, structure and usage mining. Web content mining is the process of extracting knowledge from the content of the actual web documents (text content, multimedia etc.). Web structure mining is targeting useful knowledge from the web structure, hyperlink references and so on. Web usage mining attempts to discover useful knowledge from the secondary data obtained from the interactions of the users with the web. Web usage mining has become very critical for effective web site management, creating adaptive web sites, business and support services, personalization and network traffic flow analysis.

Web usage mining also known as web log mining, aims to discover interesting and frequent user access patterns from web browsing data stored in the log files of web/proxy servers or browsers. The mined patterns can facilitate web recommendations, adaptive web sites, and personalized web search and surfing.

Various data mining techniques such as statistical analysis, association rules, clustering, classifications and sequential pattern mining

© 2005 - 2009 JATIT. All rights reserved.

have been used for mining web usage logs. Statistical techniques are the most prevalent typical extracted statistics include the most frequently accessed pages, average page viewing time, and average navigational path length. Association rule mining can be used to find related pages that are most often accessed together in a single session. Clustering is commonly used to group users with similar browsing preferences or web pages with semantically related content. Classification is similar to clustering, except that a new user (page) is classified into a pre-existing class/category off users (pages) based on profile (content). Sequential pattern mining involves identifying access sequences that are frequently exhibited across different users. All of the aforementioned techniques have been successfully deployed in various web-mining applications such as web recommendation systems, whereby web pages likely to be visited by users in the near future are recommended or pre-fetched.

Formal based Concept Analysis is a data analysis technique based on ordered lattice theory. It defines a formal context in the form of a concept lattice which is a conceptual hierarchical structure representing relationships and attributes in a particular domain. Formal concepts can then be generated and interpreted from the concept lattice using FBCA. FBCA has been applied to a wide range of domains including conceptual clustering, information retrieval, and knowledge discovery[15&16].

In this paper, we propose a novel web usage mining approach using FBCA. In particular, we are interested in mining FBCA based association rules from web logs as FBCA can be used to compute frequent patterns efficiently and accurately. In fact the number of FBCA generated association rule sis far fewer than that generated using classical Apriori-based algorithms. Moreover, the FBCA rules are comparable in quality. A side benefit is that association rules can be visualized directly from the concept lattice. These features will be illustrated and elaborated in more details in the section on performance evaluation [17].

2 FORMAL BASED CONCEPT ANALYSIS

Formal Based Concept Analysis (FBCA) [1] is a formal data analysis technique based on ordered lattice theory. It defines formal context, which is a structure that represents relationships and attributes in a domain. From the formal context, FBCA is then able to generate formal concepts and interpret the corresponding concept lattice that represents the concept hierarchy. FBCA has been applied for mining association rules and conceptual clustering of data in knowledge discovery in databases [2]. In addition, FBCA has also been applied to information retrieval [3&4].

This section discusses the basic concepts on Formal Based Concept Analysis (FBCA) [5].

Definition 2.1

A *formal context* is a triple $K = (G, M, I)$ where G is a set of *objects*, M is a set of *attributes*, and $I \subseteq G \times M$ is a binary relation between G and M. An object g in a relation I with attribute m is denoted as gIm or $(g, m) \in I$ and read as "the object g has the attribute m".

"Table 1" *A cross table of a formal context.*

	Data Mining	**Clustering**	**Fuzzy Logic**
D1	X		
D2	X	X	
D3			X

A formal context can be represented by a cross table with rows labeled by the object names and columns labeled by the attribute names. A cross in row g and column m means that the object g has the attribute m.

In Table 1, the formal context has three objects representing three documents, namely D1, D2 and D3, and three attributes representing three research topics, namely "Data Mining", "Clustering" and "Fuzzy Logic". The symbol "X" is used to indicate that the object has the corresponding attribute.

For example, document D1 has the attribute "Data Mining", which implies that the document D1 belongs to the research topic "Data Mining".

Definition 2.2

Given a formal context (G, M, I), we define $A' = \{m \in M \mid \forall g \in A: (g,m) \in I\}$ for a set $A \subseteq G$ (the set of attributes common to the objects in A) and $B' = \{g \in G \mid \forall m \in B: (g, m) \in I\}$

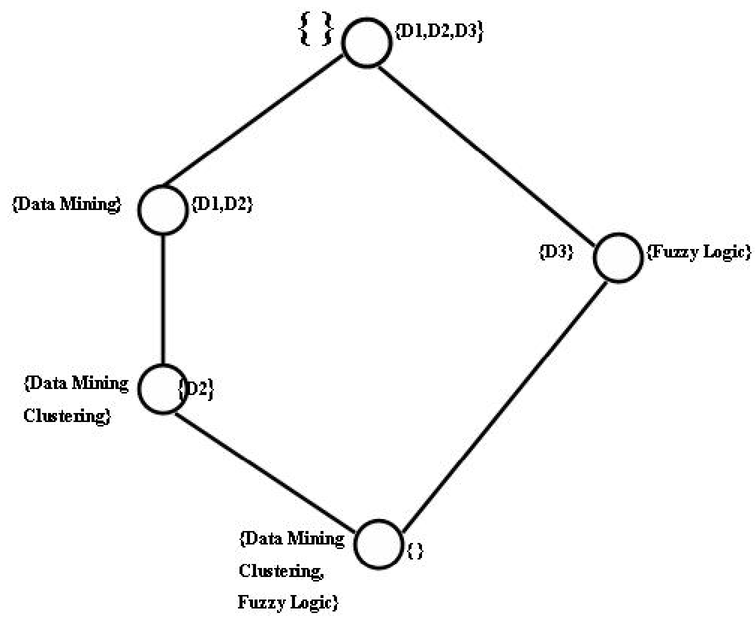
© 2005 - 2009 JATIT. All rights reserved.

for a set $B \subseteq M$ (the set of objects which have all attributes in B).

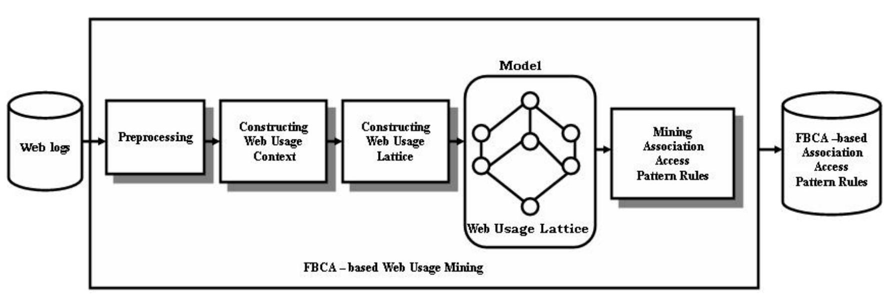
Definition 2.3
A *formal concept* of a formal context (G, M, I) is a pair (A, B) with $A \subseteq G$, $B \subseteq M$, $A' = B$ and $B' = A$. The sets A and B are called the *extent* and *intent* of the formal concept (A, B) respectively.

Definition 2.4 Let ($A1$, $B1$) and ($A2$, $B2$) be two formal concepts of a formal context (G, M, I), ($A1$, $B1$) is called the *sub concept* of ($A2$, $B2$) and denoted as ($A1$, $B1$) £ ($A2$, $B2$), if and only if $A1 \subseteq A2$ ($\Leftrightarrow B2 \subseteq B1$). Equivalently, ($A2$, $B2$) is called the *super concept* of ($A1$,$B1$). The relation £ is called the *hierarchical order* (or simply *order*) of the formal concepts. The set of all formal concepts of (G, M, I) ordered in this way is called the *concept lattice* of the formal context (G, M, I) which is denoted as $\Re$(G,M,I).

The concept lattice of the formal context in Table 1 is given in Figure 1. This concept lattice generated from the given formal context is a complete lattice, with one concept as its lower bound called *minimum* and one concept as its upper bound called *super mum*. Figure 1 also shows the intent (given at the left of each node) and the extent (given at the right of each node) of every concept. For example, the intent and extent of the concept at the top left node are {Data Mining} and {D1, D2} respectively.



"Figure 1" *The concept lattice of the formal context in Table 1.*



"Figure 2" Overview of the FBCA-based web usage mining approach.

© 2005 - 2009 JATIT. All rights reserved.

3 FBCA-BASED WEB USAGE MINING

Figure 2 gives an overview of the proposed FBCA-based web usage mining technique for association access patterns. The proposed approach consists of the following steps:

(1) Preprocessing;
(2) Constructing Web Usage Context;
(3) Constructing Web Usage Lattice; and
(4) Mining Association Access Pattern Rules from the Web Usage Lattice.

3.1 Preprocessing

The preprocessing step aims to preprocess the original web logs to identify all web access sessions. For web server logs, all users' access activities of a website are recorded by the Web server of the website. Each user access record contains the client IP address, request time, requested URL, HTTP status code, etc. Users are treated as anonymous since the IP addresses are not mapped to any user-identifiable profile database.

As discussed before, web logs can be regarded as a collection of sequences of access events from one user or session in timestamp ascending order. Preprocessing tasks [6] including data cleaning, user identification, and session identification can be applied to the original web
Log files to obtain all web access sessions.

3.2 Constructing Web Usage Context

This step will construct the web usage context (the formal context of web logs) based on all web access sessions obtained from the preprocessing step. Let M be a set of unique access events, which represents web resources accessed by users, i.e. web pages, URLs, or topics. A web access session can be denoted as $S = \{(e1, t1), (e2, t2), \ldots, (en, tn)\}$, Where $e \in M$ and ti is the request time of ei for $1 \leq I \leq n$. Note that it is not necessary that $ei \neq ej$ for $i \neq j$ in S, that is repeat of items is allowed.

For the web usage context (G, M, I), the set of objects G consists of all web access sessions in web logs for a website, and the attribute set M consists of all web resources in the website. The relation I indicates the web resources of the website that are accessed by the sessions. If $(g, m) \in I$, it says that the user in web access session g has accessed the content of the web resource m. However, it is important to note that it is not necessary mean that all web pages accessed by the user are of interest to him, as some intermediate pages might need to be accessed first before reaching the target web pages. As such, we set a duration threshold $dmin$ as a constraint to filter out non-targeted access events. The duration di of each access event ei is simply estimated as $di = (ti+1 - ti)$. Obviously, the last access event en in each session has no "tn+1" for estimating the duration. There are two ways to tackle this problem. The first way is to discard this event, and it is unacceptable, as useful data will be lost. The second way is to estimate the duration. Here, we have used the average duration in the relevant session as the estimated duration for the last access event, i.e. $dn = (d1+d2+ \ldots + dn-1)/(n-1)$. Then, all access events whose durations are less than the predefined duration threshold d $dmin$ are regarded as not useful and are discarded.

After this step, the web usage context is constructed. Table 2 shows an example of a web usage context. Each row represents a web access session(S), and each column represents a web page access(PA).

***"Table 2"** An example Web Usage Context*

	PA1	PA2	PA3	PA4	PA5	PA6
S1	X	X			X	X
S2		X	X			X
S3			X	X		X
S4	X	X	X	X		X
S5			X	X		

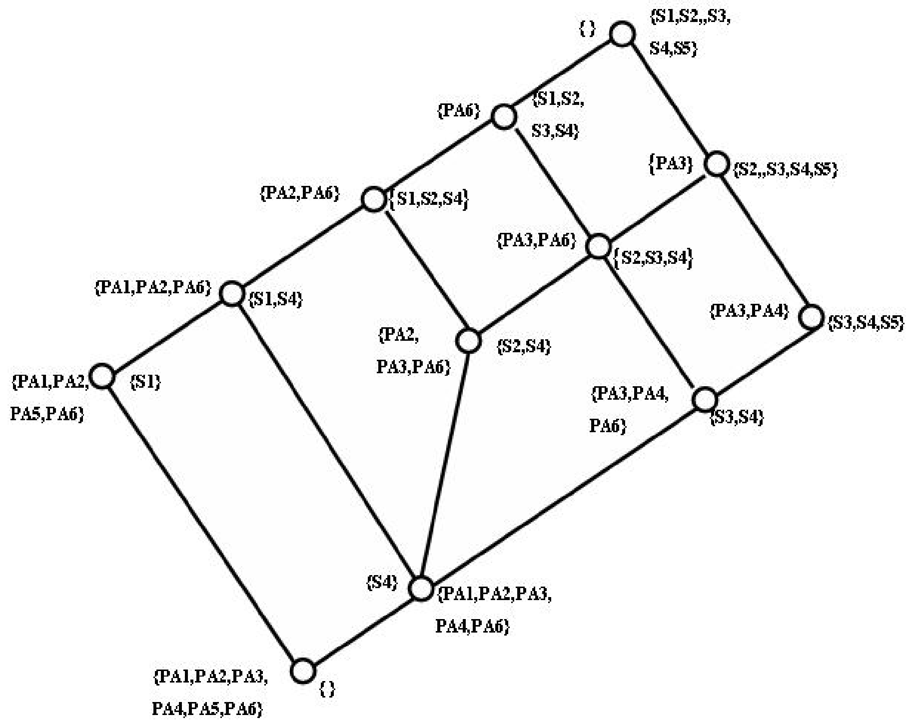
3.3 Constructing Web Usage Lattice

Based on the web usage context, a concept lattice can be constructed. We refer this concept lattice
to as *Web Usage Lattice* for the web logs. Some efficient approaches [7&8] for computing concept lattice from formal context have been proposed, which can be used for constructing the web usage lattice from the web usage context. In addition, incremental approaches [9&10] can also be used for building the web usage lattice incrementally. In this research, the TITANIC algorithm [7] is adopted for constructing the web usage lattice. It is based on data mining techniques with a level-wise

© 2005 - 2009 JATIT. All rights reserved.

approach and is much more efficient than other lattice construction algorithms, especially for weakly correlated data with a large number of objects.

Figure 3 shows the web usage lattice of the web usage context given in Table 2. The web usage lattice can be treated as a conceptual model of web logs which can then be processed by the FBCA-based technique for discovering interesting and frequent user access patterns from web usage data. The next section will discuss the mining of association rules based on the web usage lattice.



"Figure 3" *The web usage lattice of the web usage context in Table 2.*

4 MINING FCA-BASED ASSOCIATION ACCESS PATTERN RULES

Association rule mining [11] searches for interesting relationships among items in a given data set. Given a set of *items* $I = \{I1, I2, \ldots, Im\}$ and A database of transactions $D = \{t1, t2, \ldots, tn\}$ Where $ti = \{Ii1, Ii2, \ldots, Iik\}$ and $Iij \in I$, an *association rule* is an implication of the form $X \Rightarrow Y$ where $X, Y \subset I$ are sets of items called *itemsets* and $X \cap Y = \emptyset$. X is called the *antecedent* and Y is called the *consequent*. We are generally not interested in all implications but only those that are important. Here, two features called *support* and *confidence* are commonly used to measure the importance of association rules.

The *support* for an association rule $X \Rightarrow Y$ is the percentage of transactions in the database that contain $X \cup Y$. The *confidence* for an association rule $X \Rightarrow Y$ is the ratio of the number of transactions that contain $X \cup Y$ to the number of transactions that contain X .Rules that satisfy both a minimum support threshold (*MinSup*) and a minimum confidence threshold (*MinConf*) are called *strong* rules.

Association access pattern rules can be discovered based on Formal Concept Analysis. They are defined as follows.

Definition 4.1

Let M be a set of attributes of a formal context $K = (G, M, I)$. An association rule is a pair $X \Rightarrow Y$ with $X, Y \subseteq M$. Its *support* is defined by

© 2005 - 2009 JATIT. All rights reserved.

$sup\ (X => Y) = \frac{|(X \cup Y)'|}{|G|}$ and its confidence

by $conf\ (X => Y) = \frac{|(X \cup Y)'|}{|X'|}$

Let $B \subseteq M$ and $MinSup \in [0, 1]$. The *support* of the attribute set (also called *itemset*) B in formal context $K = (G, M, I)$ is $sup\ (B) = \frac{|B'|}{|G|}$

B is said to be a *frequent* attribute set if $sup(B) \geq MinSup$. A concept is called *frequent concept* if its intent is *frequent*.

Most traditional association rule mining algorithms employ a support-confidence framework. However, such approaches suffer from the same problem in which a large number of rules are usually returned. In fact, there are many redundancies among the returned association rules. In other words, despite using the minimum support and confidence thresholds to help weed out or exclude the exploration of uninteresting rules, many rules that are not interesting may still be produced. Mining association rules using Formal Concept Analysis can significantly reduce the number of rules without compromising on the quality. This approach extracts only a small subset from all association rules, called *basis*, from which all other rules can be derived.

In [12&13] different bases for association rules were introduced, which prune redundant rules, but from which all valid rules can still be derived. The computation of the bases does not require all frequent itemsets, but only frequent concept intents. In this research, the FBCA-based association access pattern rules generated for web usage mining are based on the Duquenne-Guigues basis [12] for exact association rules (i.e., association rules with 100% confidence) and the Luxenburger basis [12] for approximate association rules (i.e., association rules with less than 100% confidence). We define FBCA-based association access pattern rules as follows.

Definition 4.3

Given a formal context $K = (G, M, I)$, *FBCA-based association access pattern rules* consist of two kinds of rules:
(1) FBCA-based exact rules $B1 => B2$, where $B1$ and $B2$ are frequent nonempty concept intents, and the concept $(B1', B1)$ has concept $(B2', B2)$ as its only immediate super concept;
(2) FBCA-based approximate rules $B1 =>$ B2, where $B1$ and $B2$ are frequent nonempty concept intents, and the concept $(B1', B1)$ is an immediate subconcept of $(B2', B2)$.

Given the minimum support $MinSup$ and minimum confidence $MinConf$, for all rules $B1 => B2$ $(B1, B2 \neq \emptyset)$, $sup(B1 => B2) \geq MinSup$ and $conf(B1 => B2) \geq MinConf$.

From the above definition, it is obvious that each FBCA-based exact rule corresponds exactly to one edge that connects the subconcept with its only super concept in a concept lattice, and each FBCA-based approximate rule corresponds exactly to one edge that connects the super concept with one of its sub concepts in the concept lattice.

For example, in the web usage lattice shown in Figure.3, the edge from the concept node {PA2, PA6} to {PA6} represents an exact rule PA2=>PA6 with support = 60% and confidence = 100%, and the edge from the concept node {PA6}to {PA2, PA6} represents an approximate rule PA6 =>PA2 with support= 60% and confidence = 75%. The computation for the support and confidence is given in Figure 4.

Several approaches [12&13] have been proposed for mining association rules based on Formal Concept Analysis. Most of them compute association rules from a formal context directly. Since the web usage lattice has been constructed as a model of web logs, we propose an approach for mining association access pattern rules directly from the web usage lattice. The algorithm for mining FBCA-based association access pattern rules from the web usage lattice is given in Figure 4.

Using the algorithm given in Figure 4, all FBCA-based association access pattern rules which are given in Figure 3 can be mined from the web usage lattice. Table 3 shows the mining results with $MinSup$ = 40% and $MinConf$ = 50%. A total of 12 FBCA-based association

© 2005 - 2009 JATIT. All rights reserved.

access pattern rules including 3 exact rules and 9 approximate rules are generated.

Algorithm: FBCA-based Association Access Pattern Rule Mining

Input:

1: *WUL* – Web Usage Lattice, which is based on formal context $K = (G, M, I)$, the object set G consists of all user access sessions in web logs, and the attribute set M consists of all web pages. The relation I indicates the web pages that are accessed by the access sessions.
2: $NL = \{N1, N2, \ldots, Nm\}$ – a set of concept nodes in *WUL*, where $Ni = \langle Ai, Bi, Pi \rangle$, $Ai \subseteq G$ is the extent of Ni , $Bi \subseteq M$ is the intent of Ni, $Pi = \{Ni1, Ni2, \ldots, Nip\} \subseteq NL$ is the immediate parent nodes of Ni.
3: *MinSup* – minimum support threshold.
4: *MinConf* – minimum confidence threshold.
Output:
1: $ARS = \{AR1, AR2, \ldots, ARn\}$ – a set of FBCA-based association access pattern rules,
where $ARi = (Xi \Rightarrow Yi, Support, Confidence)$, Xi, $Yi \subseteq M$, and $Xi \cap Yi = \emptyset$.

Process:
1: Initialize $ARS = \emptyset$.
2: For each $Ni \in NL$, if $PAi = \emptyset$ and $Sup = |Ai| / |G| \geq MinSup$, do
a) If $| PAi | = 1$ and $Bi1 \neq \emptyset$, do
Insert $((Bi - Bi1) \Rightarrow Bi1, Sup, 100\%)$ into *ARS* as a FBCA-based exact rule.
b) For each $Nij \in PAi$, if $Bij \neq \emptyset$ and $Conf = |Ai| / |Aij| \geq MinConf$, do
Insert $(Bij \Rightarrow (Bi - Bij), Sup, Conf)$ into *ARS* as a FBCA-based approximate rule.
3: Return *ARS*.

"Figure 4" *The algorithm for mining FBCA-based association access pattern rules.*
For comparison, we also mine all conventional association rules using the traditional Apriori-based algorithm [14] from the web usage context given in Table 2.

The results are listed in Table 4. A total of 32 rules are generated. Among them, there are 9 exact rules (with 100% confidence) and 23 approximate rules (less than 100% confidence). It is important to note that the number of FBCA-based association access pattern rules is small compared with the number of Apriori-based association rules.

5. PERFORMANCE EVALUATION

In this section, we compare the performance of the FCA mined rules with that of the Apriori-based algorithm on web recommendation.

5.1 Web Recommendation

web recommendation systems based on web usage mining techniques. The goal of web recommendation is to determine which web pages are more likely to be accessed next by the current user in the near future. One of the motivations of building the web recommendation system is that it can be used to study the performance of the different web usage mining techniques. Association rules are well suited for web recommendations. The recommendation engine generates recommendation (links) by matching the users recent browsing history against the discovered association rules. Here ,we define three measures to evaluate the performance of FCA versus Apriori objectively.

*"**Table 3**" FBCA-based association access pattern rules mined from the web usage lattice in Figure 4.3 (MinSup = 40%, MinConf = 50%).*

No	FBCA-based Association Rules	Support	Confidence
1	*PA2 => PA 6*	*60%*	*100%*
2	*PA 4 => PA 3*	*60%*	*100%*
3	*PA 1 => PA2^PA6*	*40%*	*100%*
4	*PA 6 => PA 2*	*60%*	*75%*
5	*PA 6 => PA 3*	*60%*	*75%*
6	*PA 3 => PA 6*	*60%*	*75%*
7	*PA 3 => PA 4*	*60%*	*75%*
8	*PA 2 ^ PA 6 => PA 1*	*40%*	*67%*

© 2005 - 2009 JATIT. All rights reserved.

9	*PA 2 ^ PA 6 => PA 3*	*40%*	*67%*
10	*PA 3 ^ PA 6 => PA 2*	*40%*	*67%*
11	PA 3 ^ PA 6 => PA 4	40%	67%
12	PA 3 ^ PA 4 => PA 6	40%	67%

5. 2. Performance Evaluation

In this section, we discuss the performance evaluation of the proposed SWABRS system.

5.2.1 Evaluation Measures

The following measures are used for evaluating the performance of the proposed approach.
Let WAS = a1a2…akak+1…an be a web access sequence.
For the prefix sequence WASprefix = a1a2…ak (k ³ MinLength), we generate a recommendation rule RR = {e1, e2, …, em} using the Pattern-tree, where all events are ordered by their support.

- If ak+1 Î RR, we label the recommendation rule as correct, and incorrect otherwise.
- If there exists ai Î RR (k+1 ≤ i ≤ k+1+m, m > 0), we deem the recommendation rule as m-step satisfactory. Otherwise, we label it as m-step unsatisfactory.

Let R = {RR_1, RR_2, …, RRn} be a set of recommendation rules, where RR_i ($1 \le i \le n$) is a recommendation rule.
|R| = n is the total number of recommendation rules in R.

Definition 5.2.2

Let Rc be the subset of R that consists of all correct recommendation rules.
The overall web recommendation precision is defined as

$$\text{Precision} = \frac{|Rc|}{|R|}$$

This precision measures how probable a user will access one of the recommended pages.

Definition 5.2.3

Let Rs(m) be the subset of R that consists of all m-step satisfactory recommendation rules.
The m-step satisfaction for web recommendation is defined as

$$\textit{Satisfaction (m)} = \frac{|\text{Rs (m)}|}{|R|}$$

The *m*-step satisfaction is a very important evaluation measure for web recommendation. Actually, the next web page accessed by a user may not be the target page that the user wants. In many cases, a user has to access some intermediate pages before reaching the target page. Therefore, it is not appropriate if we only use the precision measure to evaluate a web recommendation system. The *m*-step satisfaction gives the precision that the recommended pages will be accessed in the near future (within *m* steps). Clearly, the satisfaction and precision measures are equivalent for *m* = 1. In order to realistically evaluate our web recommendation system, *m* has been set with an appropriate value *m* = 5 to indicate that the recommended page should be accessed in the *near* future.

Definition 5.2.4

Let *Rn* be the subset of *R* that consists of all nonempty recommendation rules.

$$\text{Applicability} = \frac{|Rn|}{|R|}$$

The *applicability* of web recommendation is defined as As the Pattern-tree only stores web access sub-sequences accessed frequently by users (with a support of at least *MinSup*), the recommendation rules generation approach is unable to find recommended pages if the current access sequence does not include a frequent suffix sequence, in which case the generated recommendation rule is empty.

Therefore, the applicability measure gives a rough idea of how often recommendations will be generated. Some parameters such as *MinSup* in the proposed approach can influence the applicability of web recommendation. We will discuss these parameters in the next section. Generally, the smaller the *MinSup* is, the more applicable the web recommendation is. But, this comes at the expense of increased sequential

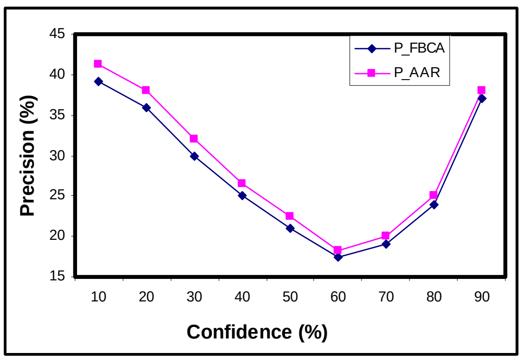
© 2005 - 2009 JATIT. All rights reserved.

pattern mining and Pattern-tree construction cost.

5.3 Performance Results

Anonymous Web Data (http://kdd.ics.uci.edu) for training and testing the web recommendation application. The training dataset for constructing the web usage lattice has a total of 5,000 sessions, with each session containing from 1 up to 35 page references from a total of 294 pages.

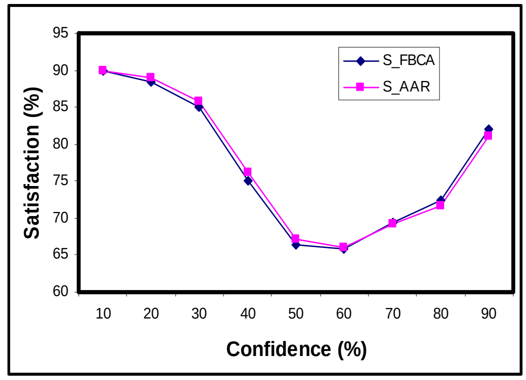
Note that we only use the 2,213 valid web access sequences that have more than two items.The test dataset has a total of 32,711 sessions and includes 8,969 valid web access sequences as the inputs to web recommendation. In the experiment, all Apriori-based association rules and FBCA-based association access pattern rules, with minimum support count as 10, are mined from the training dataset to generate recommendation rules for the testing dataset



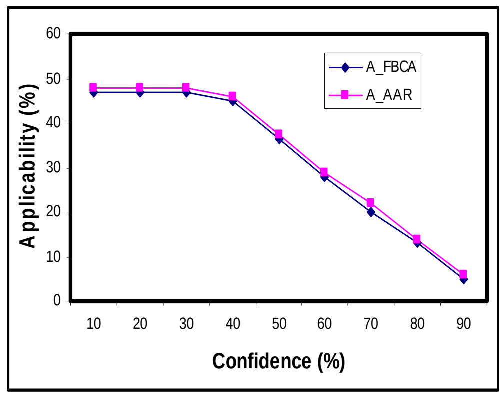
5(a)

"*Figure 5* (a,b&c)" Performance of Apriori-based association rules Vs FBCA_based association access pattern rules for web recommendation.

We use the same experimental environment that was used in *SWABRS* for measuring the performance of the *AWABRS* system. As before, we used the two datasets from Microsoft



5 (b)



5 (c)

.

6. CONCLUSIONS

We proposed a Formal Based Concept Analysis (FBCA) approach for web usage mining and evaluated it against the classical Apriori algorithm. We found the FBCA – mined rules to be 70% smaller with no noticeable performance penalty. The FBCA approach is therefore an effective and efficient tool for generating web recommendations.

REFERENCES

[1] Wille R. (1982). Restructuring Lattice Theory: An Approach Based on Hierarchies of Concepts. In I. Rival. Editor, *Ordered sets*. Boston-Dordrecht: Reidel, pp. 455-470.

[2] Hereth J., Stumme G., Wille U., and Wille R. (2000). Conceptual Knowledge Discovery and Data Analysis. In *Proceedings of InternationalConferenceon onceptualStructures*, pp. 421-437.

[3] Kosala R., and Blockeel H., (2000). Web Mining Research: A Survey. In *ACM SIGKDD Explorations*, Vol. 2, pp. 1-15.

[4] Kim M. and Compton P., (2001). Formal Concept Analysis for Domain-Specific Document Retrieval Systems. In *Proceedings of the 13th Australian Joint Conference on Artificial Intelligence (AI'01)*, Adelaide Australia, pp. 237-248.

© 2005 - 2009 JATIT. All rights reserved.

[5] Ganter B., and Wille R., (1999). Formal Concept Analysis: Mathematical Foundations. Springer, Heidelberg, 1999.

[6] Cooley R., Mobasher B., and Srivastava J. (1999). Data Preparation for Mining World Wide Web Browsing Patterns. In *Journal of Knowledge and Information Systems*, Vol. 1, No.1.

[7] Stumme G., Taouil R., Bastide Y., Pasquier N., and Lakhal L., (2002a). Computing Iceberg Concept Lattices with Titanic. In *Journal on Knowledge and Data Engineering (KDE)*, Vol. 42, No. 2, pp. 189-222.

[8] Nourine L., and Raynaud O., (1999). A Fast Algorithm for Building Lattices. In *Information Processing Letters 71*, pp. 199-204

[9] Godin R., and Missaoui R., (1994). An Incremental Concept Formation Approach for Learning from Databases. In *Theoretical Computer Science, Special Issue on Formal Methods in Databases and Software Engineering*, Vol. 133, No. 2, pp. 387-419.

[10] Valtchev P. and Missaoui R., (2001). Building Concept (Galois) Lattices from Parts: Generalizing the Incremental Methods. In *Proceedings of the 9th International Conference on Conceptual Structures (ICCS 2001)*, USA, pp. 290-303.

[11] Agrawal R., Imielinski T., and Swami A., (1993). Mining Association Rules between Sets of Items in Large Databases. In *Proceedings of the ACM SIGMOD International Conference on the Management of Data 1993*, Washington, D.C., pp. 207-216.

[12] Stumme G., Taouil R., Bastide Y., Pasquier N., and Lakhal L., (2001a). Intelligent Structuring and Reducing of Association Rules with Formal Concept Analysis. In *Proceedings of KI 2001: Advances in Artificial Intelligence*, pp. 335-350.

[13] Pasquier N., (2000). Mining Association Rules using Formal Concept Analysis. In *Working with Conceptual Structures, Contributions to ICCS 2000*, pp. 259-264.

[14] Agrawal R. and Srikant R. (1994). Fast Algorithms for Mining Association. In *Proceedings of the 20th International Conference on Very Large Data Bases*, Santiago, Chile, pp. 487-499.

[15] Stumme G., (2002). Efficient Data Mining Based on Formal Concept Analysis. In *Proceedings of Database and Expert Systems Applications (DEXA 2002)*, pp. 534-546.

[16] Stumme G., Taouil R., Bastide Y., and Lakhal L., (2001b). Conceptual Clustering with Iceberg Concept Lattices. In *Proceedings of GI-Fachgruppentreffen Maschinelles Lernen'01*, Universität Dortmund, Vol. 763.

[17] Srivastava J., Cooley R., Deshpande M., and Tan P.-N., (2000). Web Usage Mining: Discover and Applications of Usage Patterns from Web Data. In *ACM SIGKDD Explorations*, Vol. 1, No. 2, pp. 12-23.

© 2005 - 2009 JATIT. All rights reserved.

ABOUT THE AUTHORS

Mrs. D Vasumathi, had her Engineering Education in Computer Science and Engineering and Masters degree in Computer Science from JNT University Hyderabad, AP. Presently she is working as a Associate Professor in Department of Computer Science and Engineering in JNTU College of Engineering, kukatpally, Hyderabad, Andhra Pradesh. She has 9.6 years of Teaching and Administration experience. She has published number of papers at national and international journals and conferences. She is a life member of ISTE. Her areas of interest are Data Mining and Warehousing, Artificial Intelligence, Database Management Systems, Java and Web Technologies.

Dr. A. Govardhan, had his Engineering Education in Computer Science and Engineering from Osmania University and M.Tech from JNU New Delhi and Ph.D from JNT University, Hyderabad. Presently he is working as Principal at JNTU College of Engineering, Jagityala, Karimnagar (Dist), Andhra Pradesh. He has 15 years of Teaching and administration experience and has published more than 85 papers at national and international journals and conferences. He is member of many Professional bodies and received UGC fellowship award in the year 1994. His areas of interest are Data Warehousing and Mining, Information Retrieval, Computer Networks, Image Processing and Object Oriented Systems.