Journal of Theoretical and Applied Information Technology
15th November 2017. Vol.95. No 21

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

AN EFFICIENT HECTIC COMPOSITION WEB SEQUENTIAL BASED PATTERN TREES IN LARGE DATABASES

A.P.SELVA PRABHU[1], DR.T.RAVICHANDRAN[2]

[1]Ph.D Research Scholar,
Department of Computer Science,
Research and Development centre,
Bharathiar University Coimbatore and
Assistant Professor, Bharathiar University Arts and Science College Sivagiri, Erode, Tamil Nadu, India.

[2]Dean and Hod,
Electronics and Communication Engineering,
S.N.S. College of Technology,Coimbatore , Tamil Nadu, India.
[1]selvaprabhup@gmail.com, [2]dr.t.ravichandran@gmail.com

ABSTRACT

Web sequential pattern mining identifies frequent subsequences as patterns from large database. In this paper, a novel framework called Hectic Composition Mining based Approximate Pattern Tree (HCM-APT) to handle different access pattern by linking operation is presented. This framework extends the tree based indexing model, which dynamically adjusts links in the mining process using Composite Pattern Mining. A distinct feature of HCM-APT framework is that it clusters on a very limited and precisely predictable space which runs fast in memory based setting. As a result, the framework HCM-APT scales up to very large database through database segregation extensively minimizing the memory space. For dense base, competent Approximate Pattern Trees are constructed dynamically for obtaining rich properties by significantly reducing the execution time for obtaining rich properties. Finally, the proposed framework applies a scalable mining model for approximate patterns generated through tree using Variable Regression function for improving the scalability in mining large databases. Experimental results on Amazon Commerce reviews dataset show the proposed framework HCM-APT outperform other well-established methods in identifying hectic composition pattern. Experiment is conducted on factors such as execution time for obtaining rich properties, memory space consumption and scalability to mine the sequential patterns effectively.

Keywords: *Web Sequential Pattern Mining, Bidirectional Pattern, Access Patterns, Composite Pattern Mining, Approximate Pattern Trees, Variable Regression Function*

1. INTRODUCTION

Web sequential pattern mining is an significant method to investigate the access behavior of web users. It is one of the most significant and extensive problem in data mining with numerous applications, including the customer behavioral analysis over certain products, disease analysis and diagnosis, detection of crime patterns and so on.

As the Internet develops rapidly, the more peoples visits different type of websites for obtaining the information they need. Web sequential pattern mining is mostly used to recognize the behavior of web users. A web sequential pattern is kind of frequent access subsequences which assure a certain user-particular minimum support. They explains the most frequent access chronological associations of the web pages that the people visit. By applying web sequential patterns, that the users obtain more information with less operations. Besides, web sequential patterns is also help to offer personalized service for the users, which creates the users served better. Therefore, web sequential pattern mining is a better analysis and It becomes essential research of data mining. Web sequential pattern mining obtains distinguished form of patterns being generated whenever the

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

databases include lengthy sequence involving expensive space and time.

A significant shortcoming noted in conventional web sequential methods do not handle different access patterns. As a result, it failed to give accurate verification and personalized recommendation to users. In this work, an efficient framework is developed for mining the web sequential patterns.

2. LITERATURE REVIEW

Sequential pattern mining introduced in [1] considered Up Down Directed Acyclic Graph (UDDAG) approach that involved search in large spaces using Prefix Span. But this approach had the restrictions of mining for large patterns. Geographic Document Search using IR-Tree (GDS-IR) [2] worked with large high dimensional search space by constructing a tree structure.

In order to extract the hidden structures in large databases, one of the important tools is by using cluster that has been studied in a more significant manner with numerous algorithms and methods proposed. Divide and Conquer (DC) [3] approach applied a clustering algorithm based on minimum spanning tree to partition it into clusters in order to reduce the time for constructing the tree. However, the rich properties were not obtained with respect to large datasets. Another approach in [4] addressed large datasets for various social network settings through multi-objective model. The link strength between different users was not established.

A Latent Dirichlet Allocation (LDA) [5] was presented with the motive of tracking and analyzing the sentiments of various users and determining their behaviors through foreground and background topics on Twitter. A new sequential pattern mining called P-Prefix Span [6] extracted more reliable patterns using minimum time ensuring scalability.

Evolutionary measures and relationships between genes and proteins have constrained the occurrences of their sequences over generations. In [7], Multiple Sequence Alignment algorithm was designed to obtain sequences through coupled pattern mining improving patterns being generated by applying constraint-based and progressive and iterative algorithms. Though the algorithm resulted in optimized patterns being generated, but the field of research was also restricted. A novel weighted support method was introduced in [8] to address the range of applications being solved through weight assignment resulting in the improvement of time taken to execute many patterns. Generalized Sequential Pattern Mining [9] extracted more number of patterns with the aid of sequential tree from web log data. But, the method was unsuited for online application. A new algorithm called, Regular Frequent Pattern Mining in Incremental Databases [10] was designed to address online applications using vertical data format.

One of the solutions to this is the application of closed pattern mining. In [11], an efficient algorithm called, CSpan was introduced to extract the patterns in an early stage reducing the time and space required for it. However, it did not incorporated user specified constraints. To address this issue, Compact Frequent Monetary Length (CFML) [12] was designed to discover more user-centered patterns. Another method based on user specified constraints was designed in [13] using binary matrix approach that identified the frequent patterns reducing the time complexity.

One of the most challenging tasks in data mining is the extraction of the most useful and significant patterns in sequential data. Many researchers have been focusing for identifying the unique patterns for large databases. A new structure called, Co-occurrence MAP (CMAP) [14] to reduce the infrequent candidates while performing mining. However, the work lacked optimized mining. In [15], Bidirectional Growth based Cyclic Analysis of Web Patterns (BGCAWB) to optimize the generation of rules being fetched using 2-Sequence Patterns. Another method used probabilistic support [16] that extracted frequent items from databases on uncertain in nature. However, with the scalability of items generated, storage space also increased in a greater amount.

Web usage mining is one of the extensive applications of sequential mining that accesses web log files, where the web access registered by numerous users are recorded in the web server at several intervals of time. An algorithm called, Apriori All Set algorithm was designed in [17] with the objective of deriving the user reaction through traditional set theory. Though temporal patterns were included, but the time factor was not considered. Temporal sequential pattern [18] with respect to time was designed based on the active

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

statistical techniques to improve the time taken for deriving the user behavior patterns. Pattern based web mining [19] opened up the methods for efficient pruning of patterns using hypothetical testing. Personalized sequential pattern mining [20] designed with the objective of identifying the user relevance patterns at relatively lesser amount of time.

The main objective of the research work is to analysis both computationally efficient in terms of memory space, handles different patterns and competent with the state-of-the-art pattern mining techniques. Initially, Hectic Composite Pattern Mining is applied to structure the link table and minimize the memory space. Secondly, Approximate Pattern tree is constructed to minimize the execution time for obtaining rich properties. Finally, Variable Regression function is applied with Orthogonal Polynomial Regression for improving the scalability.

The rest of this paper is organized as follows: The composite based pattern mining model which includes creation of link table, construction of competent approximate pattern trees and building variable regression function is presented in Section 3. Experimental results are presented in Section 4. The last section 5 summarizes the conclusions.

3. FRAMEWORK FOR HECTIC COMPOSITION MINING BASED APPROXIMATE PATTERN TREE

This section describes the proposed framework HCM-APT which stands for Hectic Composition Mining based Approximate Pattern Tree (HCM-APT). The idea is to create a composite pattern mining where features are highly vary using Approximate Pattern tree that handles different access pattern by dynamically adjusting the link and minimizes the execution time for obtaining rich properties improving the scalability. The architectural diagram for HCM-APT is shown in Figure 1.

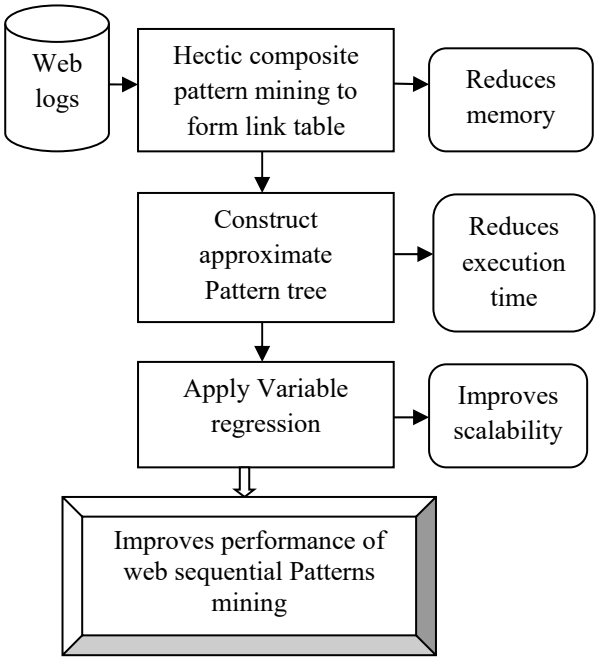

Figure 1: Architecture diagram of Hectic Composition Mining based Approximate Pattern Tree

Figure 1 shows the architecture diagram of Hectic Composition Mining based Approximate Pattern Tree. As shown in the figure, the web log database is considered as input source. From the web log database, the users who access the similar kind of web pages are mined and recognize the user behavior. The proposed HCM-APT is divided into three stages. In first stage, with the objective of reducing the memory space, composite patterns are mined by adjusting dynamic links using link table though which the composite patterns are obtained. For dense database, approximate pattern tree is constructed using composite patterns that significantly reduce the execution time with the objective of obtaining rich properties. Finally, using the approximate patterns generated, variable regression is applied to improve the scalability factor to handle different access patterns. As a Result, the performance of web sequential pattern mining is significantly improved through access pattern handling. The detailed discussion of the three stages involved in the design of the framework, HCM-APT is described in the forthcoming sections.

3.1 Hectic Composite Pattern Mining Model for Dynamically Adjusting Links (reduces memory space consumption)

Web mining is one of the types of data mining as the process of discovery and analysis of

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

practical information from the data corresponds to World Wide Web. The task of dynamically adjusting the links in web mining is performed using Hectic Composite Pattern Mining. As an exploratory data analysis tool, cluster analysis aims at grouping objects of comparable kind into their respective categories. The interesting movements or pattern identification in a large database directs to effective analysis of data. In this section, we formulate the problem of mining hectic composite pattern. Figure 2 shows the simple illustration of Hectic Composite Pattern Mining.

We propose an efficient model for composite pattern mining by adjusting the links to tackle problems of previous sequential traversal pattern mining. The main goal in this framework is to adjust the links into the composite pattern algorithm while keeping the hyperlink property. Hyperlink is patterns that the reader directly tracks either by clicking, tapping and hanging. The first step in the composite pattern mining consists of obtaining the most relevant Web data from the web log database, which is then used as the model to provide significant information about its behavior. Each set of services is involved through an execution pattern.

To obtain a composition pattern, set of services that are associated with each other are identified whose occurrences are frequently observed in several execution patterns. Such patterns which are frequently associated or linked with each others are said to form composite pattern are identified from web logs. In our model, a link table is defined, where each link table includes four vital information. The vital information included in the link table for composite pattern mining are pattern id (P_id), pattern traversal (P_T), pattern support (P_S) and pattern link (P_L) and their values are shown in table 1 and table 2 respectively.

Let us consider five patterns '$p, q, r, s\ and\ v$'.Each patterns are linked with each other through pattern link. Table 1 shows the pattern traversal and pattern link for five different patterns whereas Table 2 includes the five patterns with their corresponding support values.

Table 1: A Composite Database as an example

Pattern id (P_id)	Pattern traversal (P_T)	Pattern link (P_L)
50	q, r, s, v	q[50]⟶q[100]⟶q[200], r[50]⟶r[100]⟶r[150]⟶r[200], s[50]⟶s[100]⟶s[150],v[50]⟶v[150]
100	p, q, r, s	p[100]⟶p[150]⟶p[200], q[100]⟶q[200], r[100]⟶r[150]⟶r[200], s[100]⟶s[150]
150	p, r, s, v	p[150]⟶p[200],r[150]⟶r[200], s[150],v[150]
200	p, q, r	…….

Table 2: Composite Patterns with Support Value

P No (Pattern numbers)	P_S (Support)
1	p⟶2
2	q⟶3
3	r⟶4
4	s⟶3
5	v⟶1

Once the patterns are extracted from the web log files and are loaded into the memory, then the predictable space required for the patterns '$p, q, r, s\ and\ v$'are grouped together in a single cluster. This predictable space runs fast in memory based setting. Then, those with the similar predictable space are dynamically linked together. The design of Hectic Composite Pattern Mining with dynamic links is shown in figure 2. The entries in the link table 'L' act as the root link in a queue. From the figure, the entry for pattern 'p' in the link table 'L' is the starting point or the head of the '$p\ queue$', which dynamically with transactions 100, 150, and 200. These three composites has 'p' as their first pattern item.

In a similar manner, the pattern 'q'in transaction 50 is dynamically linked as a $p\ queue$'. But, 'r', 's' and 'v' are empty as it does not have any frequent pattern that starts with this patterns. In a similar manner, HCPM is scaled up to very large databases by database segregation in an efficient manner. As a result, the memory space consumption is significantly reduced in an extensive manner.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

3.2 Construction of Competent Approximate Pattern Trees (reduces the execution time for obtaining rich properties)

The design of Hectic Composite Pattern Mining is said to be efficient whenever the frequent patterns of a composite data base as well as the links tables sufficiently accommodates into the main memory. In case, if both do not fit into the main memory, then database segregation is performed. The framework, HCM-APT performs database segregation using competent Approximate Pattern CAP-Tree.

In this section, a data structure called, competent Approximate Pattern CAP-Tree is constructed. To minimize the enormous amount of composite patterns generated, when the data base becomes dense, as part of the mining process competent, database segregation is performed by constructing AP-Tree. While sustaining high quality patterns, recent works on composite patterns have been concentrating on mining approximate set of composite patterns.

Let '$C_a^{PS}(CE)$' represents the set of transactions in Composite Data Base 'CDB' which contain composite pattern set 'PS', segregated into 'a' parts in memory based setting with 'CE' competent error. In addition, let 'CE_{i+1}' and 'CE_{i+2}' denote two non-negative competent errors. If a transaction includes the pattern set 'PS' with 'CE_{i+1}' competent errors, then it is not possible that the transaction includes the pattern set 'PS' with 'CE_{i+2}' errors concurrently. According to this property, the approximate support of a pattern set 'PS' in a composite transaction database CDB is obtained by the following function:

$$if\ \alpha_{PS} = 0, SUP_P\ (PS) = C_a^{PS}(0) \quad (1)$$
$$if\ \alpha_{PS} = PS, SUP_P\ (PS) = CDB \quad (2)$$

With the aid of competent Approximate Pattern Tree (AP-Tree) all the patterns from Composite Data Base which includes pattern 'p' in pattern set 'PS' are initiated from the root node. The procedure for competent Approximate Pattern (AP) is given below:

Procedure AP (X, Y,R)
Step1: Scan composite Data base CDB once.
Step2: Segregation of CDB into parts '$CDB_1, CDB_2, .., CDB_a$', where '$1 \leq PS \leq a$', then the composite patterns in CDB can be held in the main memory.
Step 3: Let FPS_idenotes the composite pattern set in CDB_{PS}
Step 4: Let $FPS = \sum_{i=1}^{a} FPS_i$
Step 5: Scan composite data base CDB and extract the support for patterns in FPS
Step 6: For PS = 1 to$a$, use α_{PS} in CDB_{PS} with respect to the minimum support threshold 'min $\_SUP$' by applying (1) and (2).
Step 7: Output patterns that satisfies minimum support value 'min $\_SUP$'

Algorithm 1: Construction of competent Approximate Pattern

The above algorithm shows the composite approximate patterns being generated using composite data base. For dense database, competent Approximate Pattern Trees segregates the composite data base into different parts in such a way that it can be held in the main memory. Composite database is scanned and composite pattern set is extracted dynamically for obtaining rich properties. This application of composite Approximate Pattern Tree (AP-Tree) in HCM-APT significantly reduces the execution time for obtaining rich properties.

3.3 Variable Regression Function

Finally, a scalable mining model is applied for approximate patterns generated through tree using Variable (i.e., Orthogonal) Regression function with the aid of Orthogonal Polynomial Regression when the patterns are non-linear or scalable in nature. The Orthogonal Polynomial Regression is formalized as given below

$$PS_k = p_0(PS) + p_1\ (PS) + \cdots + p_n(PS) + CE \quad (3)$$

From (3), 'PS'denotes the pattern sets for different patterns 'p_0', 'p_1'…' p_n' and 'CE' represents the competent error.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Input: Composite Data Base, min _SUP
Output: Composite patterns mined
Step 1: Scan composite Data base CDB
Step 2: Select frequent patterns from web logs using Hectic Composite Pattern Mining
Step 3: Construct Link Table L with head, transactions, support and link
Step 4: For i = 1 to n do
Step 5: Call AP (FPS, PS, L)
Step 6: Repeat
Step 7: Travers the 'p $queue$'in the link table 'L'
Step 8: For each frequent pattern 'p'
Step 9: Construct Orthogonal Polynomial Regression using (3)
Step 10: Perform dynamic link p'to the 'p $queue$' in the link table header table 'L'
Step 11: Until composite patterns are obtained

Algorithm 2: Composite Memory-based Mining

The composite memory-based mining shows the algorithmic steps in the design of HCM-APT. The first step in the design of HCM-APT is composite database are scanned. From the web logs, frequent patterns are extracted. With the obtained frequent patterns, a link table is formed that includes the transaction number, link header and link. With this link table, composite pattern set is extracted dynamically for obtaining rich properties. In case of dense data set, the database is segregated into different parts so that the composite database is accommodated or fitted in the main memory. With the aid of minimum support threshold value, patterns are extracted. For each pattern, to address scalability of patterns being obtained, Orthogonal Polynomial Regression is applied. The above processing steps are repeated until all the composite patterns are obtained. As a result, the numbers of sequential access patterns are mined effectively based on user requirements.

4 EXPERIMENTAL EVALUATION

To evaluate the efficiency and scalability of the framework, HCM-APT, extensive performance study is done. In this section, we report our experimental results on the performance of HCM-APT in comparison with UpDown Directed Acyclic Graph (UDDAG) [1] and Geographic Document Search using IR-Tree (GDS-IR) [2]. It shows that HCM-APT outperforms UDDAG and GDS-IR and is efficient and highly scalable which handles different access by performing efficient linking operation. All the experiments were performed on a 466 MHz Pentium PC machine with 128 Mb main memory and 20 Gb hard disk, running Microsoft Windows/NT. HCM-APT was implemented by us using JAVA and the results were evaluated with the aid of Amazon Commerce reviews dataset extracted from UCI repository.

The Amazon Commerce reviews data used in our experiments to obtain different access patterns was collected by National Engineering Research Center for E-Learning. In this work, Amazon Commerce Website [21] from UCI is used as the main resource. The dataset is used in identifying the patterns for authorship where the features are multivariate. To examine the robustness of the patterns generated, patterns were obtained from 50 users and reviews collected for each author is 30. This paper used the database and by using the Hectic Composite Pattern Mining experiment is conducted to measure and evaluate the HCM-APT framework on the factors such as execution time for obtaining rich properties, memory space consumption and scalability.

5. DISCUSSION

Hectic Composition Mining based Approximate Pattern Tree (HCM-APT) is compared against the existing UpDown Directed Acyclic Graph (UDDAG) [1] and Geographic Document Search using IR-Tree (GDS-IR) [2].

5.1 Measure of Memory Space

The memory space required to construct composite pattern mining is the sum of all the memory consumed for all patterns. Lower the memory consumption, more efficient the method is said to be. It is measured in terms of kilo bytes (KB). The formalization of memory space consumption is given as below.

$$MS = \sum_{i=1}^{n} mem\,[PS_i] = mem[PS_1] + mem[PS_2] + \cdots + mem[PS_n] \quad (4)$$

MS (Using HCM-APT) = 50 + 40 + 30 + 20 + 20 = 160
MS (Using UDDAG) = 70 + 30 + 50 + 40 + 30 = 220
MS (Using GDS-IR) = 90 + 45 + 60 + 55 + 35 = 285

Journal of Theoretical and Applied Information Technology
15th November 2017. Vol.95. No 21

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Table 3 evaluates the memory space required to construct composite pattern mining based on number of patterns and is measured in terms of kilo bytes (KB). The memory space consumption based on patterns is measured in terms of KB with different number of probes ranging from 25 – 125 and comparison is made with the two existing schemes namely, UDDAG [1] and GDS-IR [2].

Table 3: Tabulation For Memory Space

Pattern size (KB)	Memory space (KB)		
	HCM-APT	UDDAG	GDS-IR
5	162	225	290
10	290	315	325
15	350	375	395
20	420	435	450
25	510	530	570
30	590	610	640
35	630	645	655

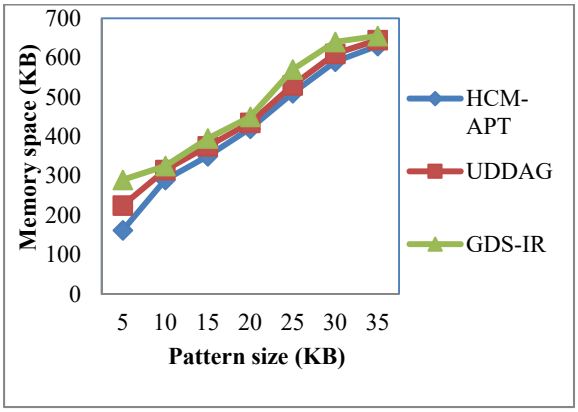

Figure 3: Pattern Size Versus Memory Space

Figure 4 shows the memory space consumption rate based on the pattern size in large databases. The memory space consumption using the proposed framework HCM-APT for each user request is measured based on the database segregation in large databases. The memory consumption is reduced using the proposed framework because of the application of Hectic Composite Pattern Mining for large databases where each set of services that are significantly associated with each other are extensively measured by evaluating the occurrences that are performed in a frequent manner. Once the patterns are extracted from web log files, then the minimum space is required for storing the multiple patterns in memory. Therefore, HCM-APT performs efficient Composite Pattern Mining to reduce memory space. The memory space is considerably reduced by 2 – 38 % compared to UDDAG [1]. Moreover based on the dynamic link, the link table establishes the pattern traversal for effectual identification of patterns which therefore help in reducing the memory space in HCM-APT by 3 – 79 % compared to GDS-IR [2] respectively.

5.2 Measure of Access Patterns Handled

The access patterns handled in HCM-APT framework is defined as the degree to which the users' requirements are met through web log files. Therefore the access patterns refer to the degree to which the patterns being mined come closer to user requirements. It is measured in terms of percentage (%). The resultant metric is:

$$AP_h = \frac{No.of\ patterns\ set\ generated}{Total\ number\ of\ patterns} * 100 (5)$$

AP (using HCM-APT) = (16/25) * 100 = 64
AP (using UDDAG) = (12/25) * 100 = 48
AP (using GDS-IR) = (10/25) * 100 = 40

The access patterns handled using HCM-APT with the existing two schemes namely UDDAG [1], and GDS-IR [2] is provided in table 4.

Table 4: Tabulation For Access Patterns Handled

Number of patterns	Access Patterns handled (%)		
	HCM-APT	UDDAG	GDS-IR
25	65	49	42
50	71	62	55
75	77	68	59
100	81	73	62
125	83	79	67
150	85	80	71
175	87	82	74

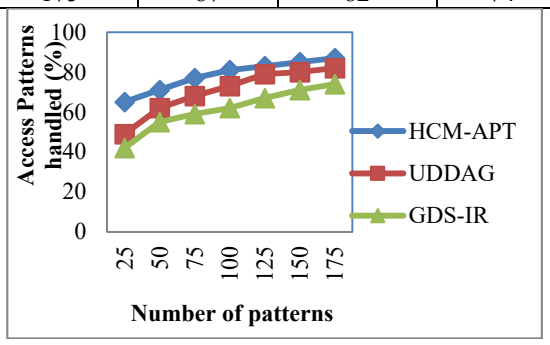

Figure 4: Number Of Patterns Versus Access Patterns Handled

Journal of Theoretical and Applied Information Technology
15th November 2017. Vol.95. No 21

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Figure 4 illustrates the access patterns handled with respect to the different number of patterns. From the figure it is illustrative that with the increase in the number of patterns, the access patterns handled is increased in all the methods. But comparatively, the access patterns handled using HCM-APT is comparatively higher than the two other methods. This is because with the application of hectic composite pattern mining, patterns extracted are loaded into the main memory and are grouped together in a single cluster. In this manner, HCM-APT different access patterns generated based on the user requirements by linking the operation through the construction of link table. Therefore, the access patterns get improved by 5 – 24 % compared to UDDAG [1]. In addition, the entries in the link table include four vital information that extracts different patterns by dynamically linking together using the support value improves the rate of patterns by 14 – 35 % compared to GDS-IR respectively.

5.3 Measure of Execution Time for obtaining Rich Properties

Execution time for obtaining rich properties is the time taken to extract the patterns from web logs. It is measured in terms of milliseconds (ms). The execution time for obtaining rich properties is given as below:

$$ET = \sum_{i=1}^{n} PS_i / n \qquad (6)$$

Where PS_i is time between user i requested for a pattern and when it is actually available and n is the total number of requests.

ET (using HCM-APT) = (9 * .25 /25) = 0.09
ET (using UDDAG) = (13 * .25/25) = 0.13
ET (using GDS-IR) = (15 * .25/25) = 0.15

The execution time for obtaining rich properties of our scheme and comparison made with two other existing schemes namely, UpDown Directed Acyclic Graph (UDDAG) [1] and Geographic Document Search using IR-Tree (GDS-IR) [2] is listed in table 5.

Table 5: Tabulation For Execution Time

Number of patterns	Execution time for obtaining properties (ms)		
	HCM-APT	UDDAG	GDS-IR
25	0.10	0.14	0.16
50	0.12	0.17	0.19
75	0.21	0.25	0.29
100	0.29	0.34	0.40
125	0.25	0.28	0.33
150	0.23	0.27	0.30
175	0.18	0.22	0.26

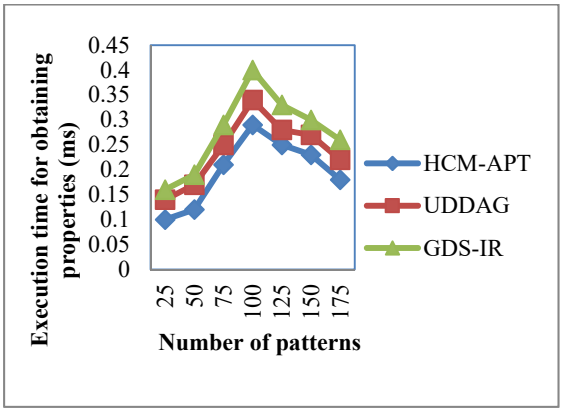

Figure 5: Number Of Patterns Versus Execution Time For Obtaining Properties

Figure 5 illustrate the execution time for obtaining properties based on the different number of patterns using Amazon Commerce reviews dataset for performing composite pattern mining for large database. Our proposed HCM-APT framework performs relatively well when compared to two other methods UDDAG [1] and GDS-IR [2]. The execution time for obtaining rich properties was significantly reduced using the HCM-APT framework because of the application of competent approximate pattern trees generated. By constructing the competent approximate pattern trees, enormous amount of composite patterns generated are minimized whenever the data set becomes dense, database segregation is performed resulting in minimizing the execution time. Composite database is scanned and composite pattern set is extracted dynamically for obtaining rich properties. From the composite pattern extraction, the support for patterns is identified with the minimum support threshold. If the output patterns are satisfied the minimum support value, then the time taken to extract the patterns from web logs is reduced. Therefore, the execution time is considerably reduced by 12 – 40 % compared to UDDAG [1]. Besides, by applying minimum

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

support threshold value, output patterns are generated using HCM-APT reducing the execution time by 32 – 60 % compared to GDS-IR [2].

5.4 Measure of Scalability

Scalability is one of the important and the most significant measures to evaluate in order to determine whether the framework, HCM-APT handle large number of patterns in a simultaneous manner. Table 6 shows the measure of scalability using HCM-APT framework and two existing methods, UDDAG [1] and GDS-IR [2] respectively.

Table 6: Tabulation for Scalability

Methods	Scalability (%)
HCM-APT	78.35
UDDAG	71.45
GDS-IR	62.35

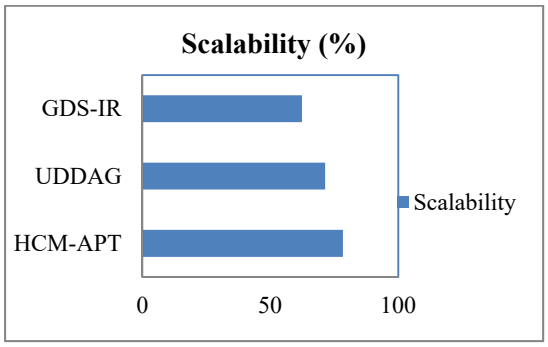

Figure 6: Measure of Scalability

Table 6 and figure 6 show the resulting scalability rate using HCM-APT, UDDAG [1] and GDS-IR [2]. From the figure it is evident that the rate of scalability is increased using HCM-APT. This is because of the application of composite pattern mining that extracts hectic features through dynamic link properties using competent approximate pattern trees. In addition, by applying Orthogonal Polynomial Regression in HCM-APT that applies a scalable mining model for mining the web sequential patterns with scalable in nature. Therefore, the scalability is improved by 9.65 % to 14.59 % compared to UDDAG and GDS-IR respectively.

6. CONCLUSION

In this work, an effective tree pattern miner using competent approximate pattern and composite web sequential pattern for large databases is mined to optimize and handles the different access patterns. The goal of Hectic Composite Pattern Mining is to extensively reduce the memory space consumption rate by constructing link table. The HCM-APT processed with three stages to improve the sequential pattern mining efficiency with respect to user requirements. Initially, the patterns are extracted from the web log files and these patterns are loaded into the memory. Therefore, the predictable space is obtained for the patterns using Hectic Composite Pattern Mining. This also helps to improve the access patterns handled. After that, approximate pattern trees are constructed to reduce the amount of time for extracting the patterns from web logs. Finally, variable regression function using orthogonal form is applied for different pattern sets to improve the scalability. Through the experiments using Amazon Commerce reviews datasets from UCI repository, we observed that our hectic composite pattern web sequential mining for large database handled different access patterns compared to existing state-of-the-art works. In addition, our composite memory-based mining algorithm effectively reduced the execution time and improved the scalability rate on several test sets. In future, the proposed work can be improved by extracting the important sequential patterns in the historical order to reduce the false positive rate.

REFERENCES

[1] Jinlin Chen, "An UpDown Directed Acyclic Graph Approach for Sequential Pattern Mining," *IEEE Transactions on Knowledge and Data Engineering*, Vol. 22, No. 7, July 2010, pp. 913 - 928

[2] Zhisheng Li, Ken C.K. Lee, Baihua Zheng, Wang-Chien Lee, Dik Lun Lee, and Xufa Wang, "IR-Tree: An Efficient Index for Geographic Document Search," *IEEE Transactions on Knowledge and Data Engineering*, Vol. 23, No. 4, April 2011, pp. 585 - 599

[3] Xiaochun Wang, Xiali Wang, and D. Mitchell Wilkes,"A Divide-and-Conquer Approach for Minimum Spanning Tree-Based Clustering," *IEEE Transactions on Knowledge and Data Engineering*, Vol. 21, No. 7, July 2009, pp. 945 - 958

[4] Christoph Dorn, Florian Skopik, Daniel Schall, Schahram Dustdar," Interaction mining and skill-dependent recommendations for multi-objective team composition", *Data & Knowledge*

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Engineering, Elsevier, Vol.70, No.10, July 2011, pp. 866-891

[5] Shulong Tan, Yang Li, Huan Sun, Ziyu Guan, Xifeng Yan, Jiajun Bu, Chun Chen, and Xiaofei He," Interpreting the Public Sentiment Variations on Twitter", *IEEE Transactions on Knowledge and Data Engineering*, Vol. 6, No. 1, September 2012, pp. 1158 - 1170

[6] Huan-Jyh Shyur, Chichang Jou, Keng Chang, "A data mining approach to discovering reliable sequential patterns", *The Journal of Systems and Software, Elsevier*, Vol.86, No.8, August 2013,pp.2196-2203

[7] K. S. M. Tozammel Hossain, Debprakash Patnaik, Srivatsan Laxman, Prateek Jain, Chris Bailey-Kellogg, and Naren Ramakrishnan, "Improved Multiple Sequence Alignments using Coupled Pattern Mining", *IEEE/ACM Transaction on Computational Biology and Bioinformatics,* Vol. 5, No. 5, Sep-Oct 2013, pp.1098-1112

[8] Gopalakrishna Kurup Raju and Achuthan Nair Rajimol," A Novel Weighted Support Method for Access Pattern Mining", *International Arab Journal of e-Technology*, Vol.3, No.4, Jun 2014, pp. 201-209

[9] Rajashree Shettar," Sequential Pattern Mining from Web Log Data", International *Journal of Engineering Science & Advanced Technology,* Vol.2, No.2, Mar-Apr 2012, pp. 204-208

[10] Vijay Kumar G, Valli Kumari V, "Incremental Mining for Regular Frequent Patterns in Vertical Format", *International Journal of Engineering and Technology*, Vol. 5, No. 2 Apr-May 2013, pp. 1506-1511

[11] V. Purushothama Raju and G.P. Saradhi Varma," Mining Closed Sequential Patterns in Large Sequence Databases", *International Journal of Database Management Systems* Vol.7, No.1, February 2015, pp.29-39

[12] Bhawna Mallick, Deepak Garg, and Preetam Singh Grover, "Constraint-Based Sequential Pattern Mining: A Pattern Growth Algorithm Incorporating Compactness, Length and Monetary", *The International Arab Journal of Information Technology*, Vol. 11, No. 1, January 2014, pp. 33-42

[13] Sowjanya Pathi, Amarendra Kothalanka, Vasudevarao Addala," Binary Matrix Approach for Mining Frequent Sequential Pattern in Large Databases", *International Journal of Advanced Research in Computer Science and Software Engineering*, Vol.4, No.12, December 2014, pp.311-317

[14] Philippe Fournier-Viger, Antonio Gomariz, Manuel Campos, and Rincy Thomas," Fast Vertical Mining of Sequential Patterns Using Co-occurrence Information", *Springer*, Jun 2014, pp. 40-52

[15] Srikantaiah K C, Krishna Kumar N, Venugopal K R1, L M Patnaik," Bidirectional Growth based Mining and Cyclic Behavior Analysis of Web Sequential Patterns", *International Journal of Data Mining & Knowledge Management Process (IJDKP),* Vol.3, No.2, March 2013, pp.49-68

[16] Radhika Ramesh Naik, Prof. J.R.Mankar," Mining Frequent Itemsets from Uncertain Databases using probabilistic support", *International Journal of Emerging Trends and Technology in Computer Science*, Vol. 2, No.2, March – April 2013, pp. 432-436

[17] Dr. Sunita Mahajan, Prajakta Pawar and Alpa Reshamwala," Analysis of Large Web Sequences using Apriori All Set Algorithm", *International Journal of Emerging Trends & Technology in Computer Science (IJETTCS)*, Vol.3, No.2, March – April 2014, pp. 292-296

[18] Naveeta Mehta, Shilpa Dang, "Temporal Sequential Pattern in Data Mining Tasks", *International Journal on Computer Science and Engineering*, Vol. 3 No. 7 July 2011, pp. 2674-2678

[19] Sheng-Tang Wu and Yuefeng Li, "Pattern-Based Web Mining Using Data Mining Techniques", *International Journal of e-Education, e-Business, e-Management and e-Learning*, Vol. 3, No. 2, April 2013, pp. 163-167

[20] Ghim-Eng Yap, Xiao-Li Li, and Philip S. Yu," Effective Next-Items Recommendation via Personalized Sequential Pattern Mining", *Springer*, May 2012, pp. 48-64

[21] Li Xiuli, Zhao Rui, and Xiao Yan," Electronic Commerce Data Mining using Rough Set and Logistic Regression", *Journal of Multimedia*, Vol. 9, No. 5, May 2014,pp. 688-693