Web Search Results Visualization Using Enhanced Branch and Bound Bookshelf Tree Incorporated with B3-Vis

S. K. Jayanthi and S. Prema, *Member, IACSIT*

***Abstract*—Enhancements in data mining for effective information retrieval is an emerging trend. This growth in turn has motivated researchers to seek new techniques for knowledge extraction. This research paper, induce the need for an incremental data mining approach based on data structure called the Bookshelf tree. The provoked approach is shown to be effective for solving problems related to efficiency of handling data updates, accuracy, processing input transactions, and answering user queries. This paper proposes a Branch and Bound Bookshelf Tree incorporated with association mining for self organization of the results retrieved from the RFDDb. This research work focus on the new techniques for keyword search over a mass of tables, and show that they can achieve substantially higher relevance than solutions based on a traditional search engine using Referenced attribute Functional Dependency Database (RFDDb). Branch and Bound is for best optimized result and the bookshelf tree is for organizing result for effective and efficient Information Retrieval (IR).B3-Vis Technique is proposed for visualizing the results retrieved from the Branch and Bound Bookshelf Tree. The relevant queries are arranged in each frame of Book Shelf for effective Information Retrieval. Finally, the search results are presented in visual mode, which allows a user to navigate between extracted schemas.**

***Index Terms*—Book Shelf Data structure, B3-VIS technique, information retrieval, referenced attribute functional dependency database (RFDDb), visualization, web-mining.**

I. Introduction

While searching the web, the user is often confronted by a great number of results, generally displayed in a list which is sorted according to the relevance of the results. Facing the limits of existing approach, this paper proposes exploration of new organizations [1] and presentations of search results, as well as new types of interactions with the results to make their exploration more intuitive and efficient. This research work is mainly focused on affording a knowledge mining tool in the form of a search engine that results list in visual mode in spite of Web page URLs as in the case of the existing conventional search engines.

Branch and Bound perform a systematic search, often taking much less time than taken by a nonsystematic search. Nonsystematic search of the space for the answer takes O (p^{2n}) time, where p is the time needed to evaluate each member of the solution space. Consider a full binary tree that has 2^n leaves. At level i the members of the solution space are partitioned by their x_i values. Members with $x_i = 1$ are in the left subtree.Members with $x_i = 0$ are in the right sub tree and could exchange roles of left and right subtree.Association mining that discovers dependencies among values of an attribute was introduced by Agrawal et al.[5] and has emerged as a prominent research area. The association mining [5] problem also referred to as the market basket problem can be formally defined as follows. Let I = $\{i_1, i_2, \ldots, i_n\}$ be a set of items. Let D = $\{t_1, t_2, \ldots, t_m\}$ be a set of transactions called the database. Each transaction in D has a unique transaction ID and contains a subset of the items in I. A rule is defined as an implication of the form $X \Rightarrow Y$ where X, Y $\subset$ I and $X \cap Y = \varnothing$. The sets of items [6] (for short itemsets) X and Y is called antecedent (left-hand-side or LHS) and consequent (right-hand-side or RHS) of the rule respectively. Several measures have been introduced to define the strength of the relationship between itemsets X and Y such as support, confidence, and interest. The definitions of these measures, from a probabilistic model is [7] given below.

I. Support ($X \Rightarrow Y$) =P(X, Y), or the percentage of transactions in the database that contain both X and Y.

II. Confidence ($X \Rightarrow Y$) =P(X, Y) / P(X), or the percentage of transactions containing Y in transactions those contain X.

III. Interest($X \Rightarrow Y$) =P(X, Y) / P(X) P(Y) represents a test of statistical independence.

Many knowledge discovery applications, such as on-line services and World Wide Web, require accurate mining information from data that changes on a regular basis [7]. In World Wide Web, every day hundreds of remote sites are created and removed. Users could be interested in finding association between keywords in web search, not necessarily satisfying the measures of the data mining rules. The main focus of this paper is the processing of the results coming from an information retrieval system. Although the relevance depends on the results quality, the effectiveness of the results processing represents an alternative way to improve the relevance for the user. RFDDb provide a schema auto-complete tool to help database designers choose a schema and to develop an attribute similarity finding tool that automatically computes pairs of schema attributes that appear to be used synonymously.

Given the current expectations this processing is composed by an organization and visualization step. This paper proposes a Branch and Bound Bookshelf Tree incorporated with association mining for self organization of

Manuscript received July 19, 2012; revised September 5, 2012.

S. K. Jayanthi is with Computer Science Department, Vellalar College for Women (Autonomous), Erode, Tamilnadu, India.

S. Prema is with Computer science Department, K.S.R. College of Arts and Science,Tiruchengode-637215, Namakkal district, Tamilnadu, India(e-mail: prema_shanmuga@yahoo.com).

10.7763/IJMLC.2012.V2.206

the results retrieved from the web search.B3-Vis Technique is proposed for visualizing the results retrieved from the Branch and Bound Bookshelf Tree.

II. Literature Review

Most previous search engine analysis research involved evaluating search engines using metadata in such areas as size, change over time, overlap, and usage patterns. Gatterbauer, et al. attempted to discover tabular structure without the HTML table tag, through cues such as onscreen data placement [3]. Chen, et al. tried to extract tables from ASCII text [2]. Penn, et al. attempted to reformat existing web information for handheld devices [4]. Alaaeldin Hafez, Jitender Deogun, and Vijay V. Raghavan [7] propose the Item-Set Tree: A Data Structure for Data Mining. Chen, et al. tried to extract tables from ASCII text [2]. Penn, et al. attempted to reformat existing web information for handheld devices [4]. Effective bias is necessary for the constraints selection in order to make it a more practical technique [8]. Web image search results clustering algorithms [9] have been proposed to cluster the top returned images using visual and textual features.

III. The B3-VIS Tree

The B3-VIS tree T is a graphical representation of the transaction data file F. Each node n ε T represents a transaction group s. All Keyword searches that are having the same semantic belong to the same clustering group. Let I={i_1, i_2,...,i_k} be an ordered set of keyword search. For two transactions n_i={a_1, a_2,...,a_l} and n_j={b_1, b_2,...,b_k}, let $n_i \leq n_j$ if and only if $x_p \leq y_p$ for all $1 \leq p \leq \min(l,k)$. Where l and k, are the lengths of n_i and n_j, respectively. Each keyword search in tree T represents either a main domain set, or a subset of main domain set in the bookshelf. Node n_i is ancestor node of node n_j, if $n_i \subset n_j$ that in n_i={a_1, a_2,...,a_l} and n_j={a_1, a_2,...,a_k}, for some l<k.Rate of recurrence of a node n is denoted by f(n).The item-set tree is constructed by transactions inserting process: The root node r represents the null item set $\varnothing$ [7]. A transaction n is inserted by examining (in order) the children of the root node r. Each time a node is inserted, f(r) is incremented by 1. The insertion process successfully ends with one of the following cases as in Fig.1.

Scenario1: In Bookshelf each rack maintains nodes *nj* which share no leading elements in *s*. When a new reference node *s* is inserted as a new shelf of *r*, *f(n)* is initiated to 1.

Scenario 2: $n=n_j$, the web search link already exists. $f(n_j)$ is incremented by 1.

Scenario 3: $n \subset n_j$, *n* is an ordered subset of book shelf node n_j. A node *n*, is inserted as a reference link of *r* and as a main domain set of n_j. $f(n)=f(n_j)+1$ and if $n_j \subset n$, node n_j is an ordered subset of *n*. The new shelf, that has n_j as a root, is reviewed and the procedure starts.

Scenario 4: $n \cap n_j \neq \varnothing$ there exists an ordered intersection between *n* and n_j. If relevance exists then the related links are placed in the same shelf of Bookshelf datastructure.A node n_i, $n_i = n \cap n_j$, is inserted between *r* and n_j, and a node *n* is inserted as a child of n_i. $f(n_i) = f(n_j)+1$, and *f(n)* is initiated to 1.

```
Algorithm B3-VIS (n,T)
n is an input keyword search
T is the B3-VIS tree
begin
  r=root (T)
  frequent occurence f(r)
  if n = keyword search (r) then exit
  choose Tn=subtree(r) such that s and keyword
          search(mainshelf (Tn)) are comparable
if Tn does not exist then
create a new shelf x for r, keyword
                        search (x) =n and f(x) =1
else if main shelf(Tn) ⊂ n then call Construct (n, Tn)
else if n ⊂ mainshelf(Tn) then
create a new node x, as a new shelf of r,
keyword search (x) =s and f(x) = f(root(Tn))+1
else create two nodes x and y, x as the
mainshelf(Tn),n.t.items(x) = n
mainshelf (Tn), f(x) = f(mainshelf (Tn))+1,
 y a new shelf of x
f(y) = 1
end
```

Fig. 1. Algorithm B3-VIS

IV. Recurrence Counting

Recurrence Counting Algorithm calculates the frequency of a keyword search *s* by adding up frequencies of those encountered item sets, which contain *s*. Recurrence counting algorithm given below in Fig. 2, demonstrates how to count frequencies of relevant documents.

```
Algorithm VisCount(s,T)
// An keyword search s, and an item-set tree T.
// Recurrence count c of item set s.

begin
r =mainshelf(T)
if s ⊂ r then c(s)=c(s)+c(r) ; end
if r<s and last-item(r)<last-item(s) do
add new node T     * new rack in bookshelf
call Viscount(s, T)
end if
end
```

Fig. 2. Algorithm VisCount

V. Referenced Attribute Functional Dependency Database (RFDDb)

The RFDDb lists each unique S found in the set of relations; along with a count that indicates how many relations contain the given S. Assume two schemas are identical if they have the same set of attributes. The RFDDb A is a set of pairs of the form (S, n), where S is a schema of a relation in R, and n is the number of relations in R that have the schema S.Extracting the RFDDb in the corpus R is a straightforward task, as described below.

For each unique schema R, the RFDDb contains a pair of the form (r, f) where f the frequency of schema r

```
Function createRFD(R)
A = {}
Viewed Domains = {}
for all r ε R
if receivedDomain(R.u)≠ ViewedDomains[R.S]
      then
   ViewedDomains[R.S].add(receivedDomain(R.u))
     A[R.S] = A[R.S] + 1
   end if
end for
```

The RFDDb is simple, but it critically allows the user to compute the probability of seeing various attributes in a schema. For example, r (marks) is simply the sum of all counts c for pairs whose schema contains marks, divided by the total sum of all counts. It also detects relationships between attribute names by conditioning an attribute's probability on the presence of a second attribute. Student staff relationship is given in Fig. 3.

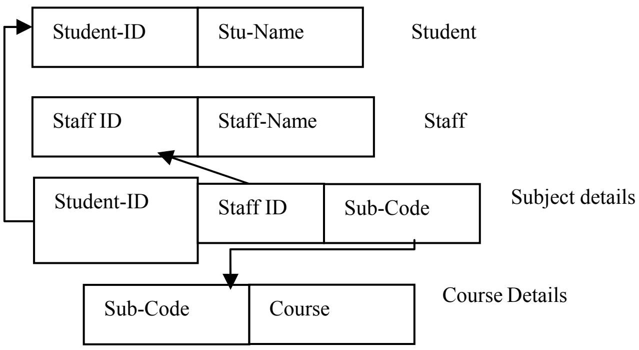

Fig. 3. Student staff relationship

VI. Bookshelf Data Structure

The Bookshelf data structure [10] as in Fig. 4. has been introduced for community formation, which stores the inverse indices of the WebPages. This data structure is formed by combining a matrix and list with dynamically allocated memory. This is an extended data structure of hash table and bi-partite core [5], which is used to store base domain and sub-domain indices of various communities. A recent study [5] shows that 81.7%of users will try a new search if they are not satisfied with the listings they find within the first 3 pages of results. However it would be too restrictive to only consider the first 30 results (10 results per page). Indeed this study has been done on search engines with linear results visualization (ordered lists) and users may want to see more results on visualizations like web graphs.

VII. Evaluation

The Result analysis of the existing web search engines like Google, Yahoo, Alta Vista and MSN will give the result in random manner based on the content and classifications is summarized in Fig.5.

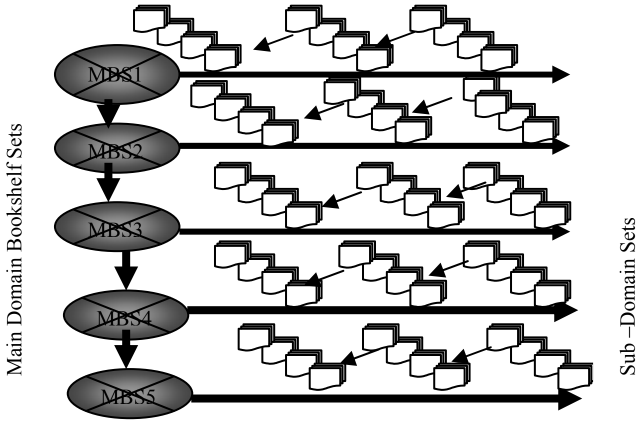

Fig. 4. Bookshelf data structure

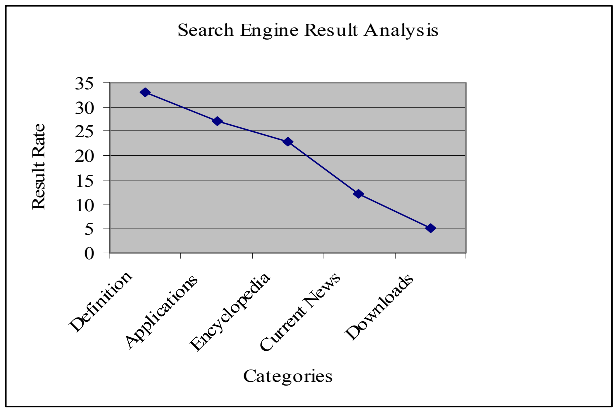

Fig. 5. Result analysis

Using Book Shelf Data Structure if the user is searching the concept, for example Operating system then the web page results is given in graph format. In which each node represents the URLs and links represent the relationship between them. The result is in visual mode in spite of web pages as in the existing system. The simulated result is given in Fig. 6.

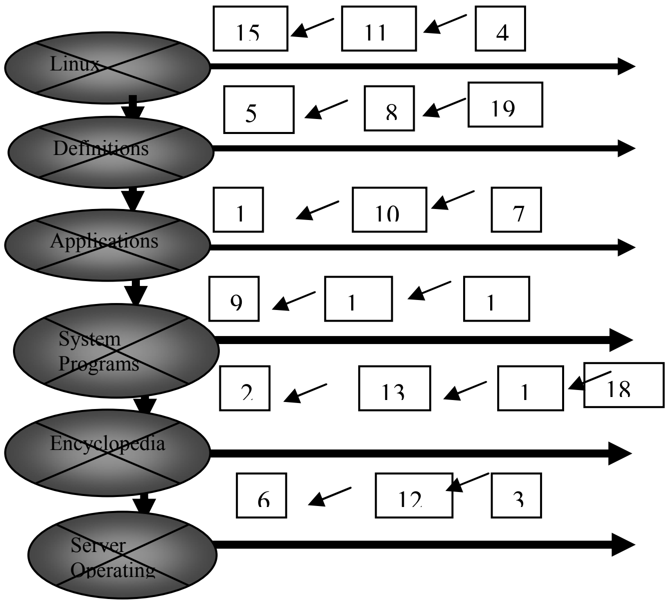

Fig. 6. Simulated result

If the related relations are recognized then the similarities can be identified using Referenced attribute Functional Dependency Database (RFDDb) [11] and presented in the each shelf of Bookshelf Data Structure (BSDS) [12][13] and the web search result is represented in web graph format as shown in Fig. 7.instead of web link format as in the existing system.

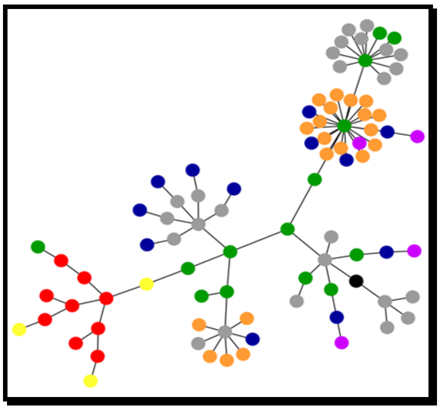

Fig. 7. Web graph simulated result of search engine results

A comparative study of web search results using BSDS (Book Shelf Data Structure) and the B3–Vis (Branch and Bound Bookshelf with Visualization Technique based on the domain operating system is given in Fig.8.If the user is searching for the specific domain (e.g. operating systems) then the related links like windows, Linux, Mac, and UNIX are arranged in Bookshelf Data Structure Format. From the graph it is clear that using B3-Vis Technique the information retrieval is efficient and effective.

VIII. Conclusions and Discussion

In this research paper, it has been introduced a new approach for association mining, called the B3-VIS tree. The new approach solves some of the problems inherent in traditional data mining techniques, such as, data updates, accuracy of data mining results, performance, and user queries. The B3-VIS tree approach maintains a structure to handle recurrence counting of transaction data, which allows future updates. B3-VIS algorithm (Branch and Bound Bookshelf Tree for Visualization of result), to insert transactions into the tree, and the recurrence counting algorithm to count the frequencies of search results has been proposed. Bookshelf Data Structure for organizing documents retrieved from Referenced attribute Functional Dependency Database (RFDDb) is defined for effective information retrieval to display the result in visual mode.

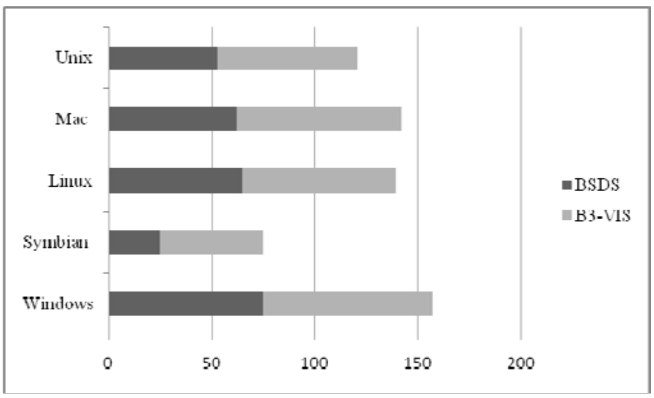

Fig. 8. Comparision of BSDS and BSDS-Vis

References

[1] Dittenbach, D. Merkl and A.Rauber, "Using Growing Hierarchical Self-Organizing Maps for Document Classification," ESANN, 2000, pp.7-12.

[2] H. Chen, S. Tsai, and J. Tsai, "Mining tables from large scale html texts," in *18th International Conference on Computational Linguistics* (COLING), 2000, pp.166–172.

[3] W. Gatterbauer, P. Bohunsky, M. Herzog, B. Kr¨upl, and B. Pollak, "Towards domain-independent information extraction from web tables," in *Proceedings of the 16th International World Wide Web Conference* (WWW 2007), 2007, pp.71–80.

[4] G. Penn, J. Hu, H. Luo, and R. McDonald, "Flexible web document analysis for delivery to narrow-bandwidth devices," in *International Conference on Document Analysisand Recognition* (ICDAR01), 2001, pp.1074–1078.

[5] R. Agrawal, T. Imilienski, and A. Swami, "Mining Association Rules between Sets of Items in Large Databases," in *Proc. of the ACM SIGMOD Int'l Conf. On Management of data*, May 1993.

[6] R. Agrawal, and R. Srikant, "Fast Algorithms for Mining Association Rules," in *Proc. Of the 20th VLDB Conference*, Santiago, Chile, 1994.

[7] Alaaeldin Hafez, Jitender Deogun, and Vijay V. Raghavan, "The Item-Set Tree -A Data Structure for Data Mining," DEPSCoR program of Advanced Research Projects Agency, Department of Defense, 2000.

[8] C. Carpineto,S.Osinski,G.Romano,and D.Weiss, "A survey of web clustering engines," *ACM Comput.* Surv. 41(3) ,2009.

[9] K. Czajkowski, S. Fitzgerald, I .Foster, and C. Kesselman, "Grid Information Services for Distributed Resource Sharing," in *10th IEEE International Symposium on High Performance Distributed Computing*, IEEE Press, New York , 2001, pp. 181--184.

[10] S. K. Jayanthi andS. Prema,(June 2011).Vis-Proclus: A Novel Profiling System for Instigating User Profiles from Search Engine Logs Based on Query Sense.International Journal of Engineering Science and Technology. pp.4564-4571. [Online]. Available: http://www.ijest.info/issue.php?file=vol0 3issue06

[11] S. Prema and S. K. Jayanthi,"Referenced Attribute Functional Dependency Database for Visualizing Web Relational Tables," in *Proc.International Conference on Network and Computer Science* (ICNCS 2011), indexed by the Ei Compendex and Thomson ISI (ISTP), IEEE Xplore ,Kanyakumari , 2011.

[12] S. Prema and S. K. Jayanthi, "Facilitating Efficient Integrated Semantic Web Search with Visualization and Data Mining Techniques," in *Proc.International Conference on Information & Communication Technologies*, Springer Transl.sep, 2010, pp. 437 – 442.

[13] S. Prema and S. K. Jayanthi, "Bookshelf Data Structure Incorporated Visualization for Efficient Integrated Semantic Web Search," in *Proc. International Conference On Emerging Trends In Mathematics And Computer Applications*, Mepco shlenk college of engineering, Dec 2010.

S. K. Jayanthi received the M.Sc., M.Phil., PGDCA, Ph.D in Computer Science from Bharathiar University in 1987, 1988, 1996 and 2007 respectively. She is currently working as an Associate Professor, Head of the Department of Computer Science in Vellalar College for Women. She secured District First Rank in SSLC under Backward Community. Her research interest includes Image Processing, Pattern Recognition and Fuzzy Systems. She has guided 18 M.Phil Scholars and currently 4 M.Phil Scholars and 4 Ph.D Scholars are pursuing their degree under her supervision. She is a member of ISTE, IEEE and Life Member of Indian Science Congress. She has published 6 papers in International Journals and one paper in National Journal and published an article in Reputed Book. She has presented 15 papers in International level Conferences/Seminars (Papers has been published in IEEE Xplore, ACEEE Search Digital library, Springer Digital Library and online digital library) in various places within India and in London (UK), Singapore and Malaysia, 16 papers in National level Conferences/Seminars and participated in around 40 Workshops/Seminars/Conferences/FDP.

S. Prema, currently working as a Assistant Professor in K.S.R. College of Arts & Science has received the B.Sc., M.C.A., M.Phil., from the Periyar University in 2001,2004,2008 respectively and now pursuing Ph.D in computer science at Bharathiar University. Her area of Doctoral research is Web mining.