IOSR Journal of Computer Engineering (IOSR-JCE)
e-ISSN: 2278-0661,p-ISSN: 2278-8727
PP 42-48
www.iosrjournals.org

Identifying Most Popular Traversed Paths inNavigation Patterns of E-Commerce

Dr. AbhaS. Khandelwal
(HOD Department of Computer Science, Hislop College, Nagpur, RTM Nagpur University, Nagpur, Maharashtra, India).

Abstract- *The growth and widespread adoption of Internet-based services and technologies are major phenomena, causing fundamental changes in the way we work and live.Businesses currently employ the Internet to facilitate a number of processes, from simple e-mail communication to complex supply-chain integration. Clearly, the Internet has reached such a degree of reliability and integration into everyday life that it is regardedas infrastructure, i.e., a service that is assumed and relied upon. The Internet is triggering tremendous growth in E-Commerce activities in recent years - in both (B2B), the Business to Business and the (B2C),Business to Consumer sectors.As the Internet is growing both in terms of number of users and usage, delivering reliable and valuable service is becoming increasingly important for e-businesses. It is very important for the e-shop to keep the user satisfied with the content that is delivered. The search optimization plays a key role in transforming a visitor into a customer.To address this problem, this work is focused on developing cost effective, scalable e-business infrastructures that can support the future growth of the E-Commerce. While designing and developing an e-business infrastructure one of the important aspects is "mining valuable web usage behavior". In this paperwidely used approachesfor discovering ''Critical edge sequences (CESs)"is studied and new approach is being suggested to identify most popular traversed path, which may find place in developing infrastructure which in turn may result inenhancing the customer base.*

Keywords: *E-Commerce, CES (Critical edge sequences), web usage behavior*

I. INTRODUCTION

As the Internet is penetratingin the day-to-day activities of life, E-Commerce isfinding its place as major services, at a massive rate. To meet this demand, organizations are investing significant amounts in building e-business infrastructures.In fact, according to a recent Gartner report [1], e-business spending is growingtwo and half times as fast as overall IT spending. If the infrastructure architecturewere designed properly, these enormous spending can be controlled."web usage behavior" Mining is vital. This can be termed as one of the valuable aspect to tackle this problem.

II. MINING VALUABLE WEB USAGE BEHAVIOR

Study of Web usage behavior is one of the prominent critical factor for a successful e-business, and this problem has attracted, least attention of researchersin view of internet infrastructure issues.When a user interacts with an e-business site, the site delivers content to the user. It is desirable for the site to keep the user encouraged and satisfied with the content that is being delivered. It plays vital role in enhancing customer base of e-commerce. In fact, study shows that, themore time a visitor spends on a site, the more money the visitor spends on the site [2].Thus, "stickiness" to a site, is an important objective for e-businesses. In order toachieve this objective, it is imperative for the site to serve content that is of interest to theuser.

Personalization is a technology that identifies a user's interests and determines, the appropriatecontent based on their interests to deliver. When a user interacts with an e-businesssite, every interaction is logged into a file on the server side. This web log file contains a varietyof information, such as what the user requested, where the request originated from (typically anIP address), and whether the request was successfully fulfilled [3]. A widely used techniqueto identify users'interests isweb mining, i.e., applying data mining techniques to web log data [4, 5, 6]. Web mining enables sites to discover interesting and useful patterns inuser behavior. Subsequent analyses of these patterns often yield valuable information forthe site.

One type of pattern discovery that is fundamental to e-business sites, is todetermine the most popular traversed paths in the site. Identifying the most popular traversed paths enables a wide variety of analyses, which may be extremely useful. In fact, if an analysis reveals that the most popular paths are consuming lot of time to traverse,orfrom server side there are delays, or large download time is required for downloading large number of images, or if the number of clicks is large, this would help in highlighting possible areas of improvement on the site. Finding popular paths can be useful in several other scenarios as well, for example, in

finding behavior of traversal, or heavily-traversed sequences may be provided preferential resource allocation on the site. Analysis of these paths can also reveal valuable patterns that can aid marketing and advertising efforts on the site, e.g. for delivering advertisements, e-coupons, or running promotional campaigns.

Analyses of the most popular traversed paths are extremely valuable to e-businesses butthe actual discovery of these patterns is very difficult to do in practice due to the sheer size of the log files. It is not uncommon for log files for e-business sites to be in the terabyte range in size. Consider, for example. Yahoo!, which serves more than millions of pages per day. Assuming each click generates, on average ten entries in the log file and each entry in the log file is about 300 bytes, the resulting log file generated for a single day will approximately bein few terabytes. Even the most efficient path finding algorithm will take days to operate on this huge volume of data. Thus, for identifying the most popular traversed paths within a reasonable time frame (e.g., within a few hours) is the need of the hour.

III. RELATED WORK

Discovering CESs

Path-finding algorithms are a staple of graph-theoretic literature.Dijkstra's Single-Source Shortest-Path (SSSP) algorithms [7] appear similar in nature to the problem of finding the most popular paths between a node pair on an E-Commerce catalog - both problems involve the need to identify paths through a graph, based on some criterion.However, the criterion for ranking paths is different in the two problems - SSSP finds the shortest paths through the site, based on a set of edge weights, where in E-Commerce scenario there is a need to find heavily-traversed sequences of edges in a graph.

At first glance, the maximization characteristic of this problem seems to suggest amapping to the longest-path problem more than shortest-path. The longest path problem seeks to determine the simple path (i.e. no repeated edges) of greatest weight in a graph G. This problem is known to be NP-complete [8] for graphs which may contain cycles (for acyclic graphs, there exists a mapping to the shortest-path problem, which is known to be in P). The difficulty in solving the longest path problem lies in verifying that adding a particular edge will not violate the simple path constraint. The NP-completeness of the longest path problem is due to the fact that adding a new edge to the path will always increase the weight.

In most graph problems, such as SSSP, minimum spanning tree and maximum flow, the edge weights are assumed to be deterministic and independent, i.e., edge weights are known in advance and the weight of an edge is not dependent on the weight of any other edge in the graph. However, in the CES-finding problem edge weights are non-deterministic and dependent, as they represent the probability of traversing an edge. Accordingly, the method proposed in this work is basically drawn from existing path-finding literature, but include their own means of handling the dependencies among edge probabilities. The work described above primarily addresses the problem of discovering CESs from the paths output. For practical values of($k< 10$)the time burden of performing this task (i.e., common substring extraction from a small set of input strings) is negligible. The experiments consumed, for $k = 10$ time in the order of single digit milliseconds. Thus, it turns out that discovering the k most traversed paths between specified node pairs is the most difficult part of the CES discovery problem.

Web Usage Mining

Another area of related work is the domain of web usage mining (e.g.. [9, 10, 11,12, 13]). Much work on algorithms for web usage mining (e.g.. [11, 14]) is based on fundamental work in data mining [15, 16 , 17,18] and sequential mining in particular [19,20]. Virtually all approaches in this class of solutions take web logs (i.e.. full data) as input, and provides as output sequential patterns exactly matching the criteria. For example, in [12] it finds all repeating subsequences meeting a minimum frequency threshold, while in [13] it suggests a document prefetching scheme based on predicting a user's next action on the site. In [21], the Web Utilization Miner (WUM) is presented. The WUM software mines web logs for patterns specified in the query language MINT. The work in [22] and [23] is also relevant. Here, the authors make use of Markov models for predicting a user's next move in a Web site. In [22], the authors use higher-order Markov models and pruning techniques (based on support, confidence, and error rates) to reduce the state space. In [23], the authors mine LRSs (longest repeating subsequences), a subset of Kth-order Markov models to predict users' next actions on a Web site. However, rather than generating exact solutions and then pruning (as suggested in [38]), or retaining only the longest repeating subsequences ([23]), these require exact information as input.In addition, both [22] and [23] attempt to predict a single step, given a prior traversal path, whereas in this work, focus is to predict CESs ofarbitrary length.

a. Widely used Approach to Discover E-commerce Catalogs and Critical Edge Sequences:

‘‘Critical edge sequences (CESs) are frequently traversed subsequences, on the traversal paths leading from a specified start node to a specified end node in a Web site graph". Specifically, these are “frequently visited sequences of pages on the traversal paths from the start page to the end page”.Web sites allow the collection of vast amount of navigational data,also known as clickstreams of usertraversals, through the site. Uncovered, interesting patterns within the dataset may be found in massive data stores. For e-businesses, looking for anedge, constantly in the tremendouslycompetitive “online shops”,may result infetching interest of researchers.

The data on user traversal is analyzed after tracking of visitors traversing through the site of bookseller say “Buybooks.com”is over. It is assumedthat the following information is the outcome of such analysis:

A visitor, in a session,navigates through Novel and Autobiography, tends to eventually make apurchase of anAutobiography book; When a user takes shorter time to traverse thesite’s between Novel and Autobiography, the higher are chances of an eventualpurchase. The knowledge of such relationships represents a potentially valuable opportunityto Buybooks.com.If the popular traversed paths betweenNovel and Autobiography can be appropriately recognize and analyze, then they could be analyzed to determine the average length of the traversaltime from Novel and Autobiography.

If such analysis reveals that the mostpopular paths were prolongedunnecessary while traversing.e.g. if there are server-side delays, or large numbersof images lead to large download times, or if the large number of clicks are required), thiswould highlight possible areas of updation on the site. Finding popular paths can beuseful in several other scenarios as well, for example in predicting traversal behavior forcaching. It may also support to provide resource allocation on preference basis,for heavily-traversed sequences on the site. Analysis of these paths can also reveal valuable patterns that can aid marketingand advertising efforts on the e-shop, e.g., for publishing advertisements, e-coupons, or runningpromotional campaigns.

To motivate the specific problem addressed in this work. Suppose that an analysis of the most frequently traversed paths betweenNovel and Autobiography found two distinct paths:

1. *Novel*—►*History*—► Science —► Mythology —►Interior—► English Literature —►*Autobiography*

2. *Novel*—► *Sociology*—►*Science* —► *Mythology*—►*Interior*—►*Education*—►*Autobiography*

In addition to identifying these most-traversed paths, it is also interesting to note thatthe most popular paths between*Novel*and*Autobiography*share a commonsubsequence, i.e.,*Science*—► *Mythology*—►*Interior*, indicating that consumercommonlytraverse this subsequence while navigating from*Novel* and*Autobiography*(regardlessof which exact path they may have followed).Such edge sequences are known as*CriticalEdge Sequences* (CESs).

The quest for valuable information from accumulated data has always been of interestto researchers [15, 16,19]. Such explorations have been carried out both from a systems perspective(e.g., in database query processing algorithms - the creation of frequency histograms for use) and from an applications perspective (e.g., the mining of association rules froma large dataset). This work belongs primarily to the lattercategory; the work is focused on,to bring upvital information from large volumes of data byidentifying critical traversal patterns at a web site,which is the main contribution in this area. In particular, given a hierarchical navigationstructure (i.e., a catalog), and a set of traversal records over this structure (i.e. user navigationsessions over the catalog), the focus is on, discovering critical edge sequences(CES) between a pair of nodes.

In this approach CES are discovered by a two step process:

a. The*k* most traversed paths are identified between the two nodes:
b. The CESs. i.e.. thecommonly occurring subsequences are discovered within these *k* paths

b. Discovering CES using Site Model

Site Model:A graph-based representation of a site, similar tothe model in [24], and concepts of user-site interaction found in [25]. A site is a graph$G = (V, E)$ where each node v $\in V$ represents a particular concept in the E-Commerce site,perhaps a product or a category of products, and each edge e = (u,*v)* $\in E$ represents thelinks on the site, i.e., the possible traverse from node u to node *v* on the site.

In this model, a particular customer’s traversal through the site can be stated as a *click sequence*. Click sequence is one, which providesa sequence of nodes, the user visited, and also the order in which user visited

them.Following [25] the model separates the (abstract) site content representation from the(concrete) page delivered to a user. Essentially, it is assumed that content associated withmultiple nodes in the site graph can be delivered to a user on a single page, and allowsimmense combinations of site content to be generated. Practically speaking, this type ofmodel maps well, a recent phenomenon in Web content delivery - customized and dynamicpage generation. Here, the pages served to a user are based on thesite's the user's state at run-time and business logic. Often, a site's business logic utilizes a user's previous navigation patterns to build newpages for a user. For example, look at the technique applied in amazon.com site. User's recently viewed products, as well as recommended products are provided in this site. For this alink "Page You Made"is provided to theservice consumer.This site clearly uses a model similar to that described in [25], where content fora page may be drawn from multiple nodes in the site graph, and the business logic thatdetermines the content of a page is dependent on the user's previous traversal through thesite.

c. Critical Edge Sequences: Session Graph

A more precise definitionis provided here:

Definition Consider a Web site graph $G = (V, E)$, consisting of a set of vertices V andedges E . Suppose we are given a given a start node (i.e., a vertex) $N_a \in V$and an endnode $Nb \in V$, such that from N_a,$N_b$is reachable. Let Z denote the set of unique paths in G from $N_a$toN_b.Let J denote the binary relation (S, F), which may be possibly partially overlapped.

where any $s_t \in S$ denotes a subpath computable from any element in$Z$ and a tuple $(s_a. f_a)$denotes that the subpaths$_a$ was traversed f_a times. In G,an edge sequenceisdenoted by$s_a$andfrequency of traversal is denoted by$f_{a,}$which is the frequency of traversal of$s_{a,}$the edge sequence.CES of $G$is any edge sequencein G whose frequency of traversal is greater than some predetermined threshold frequency$T$.

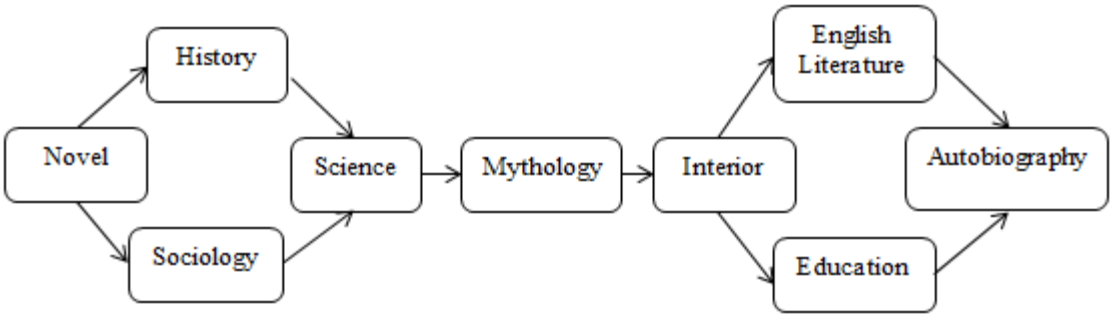

Fig 1. Session Graph

To clarify this notion,let us discoverCritical edge sequences (CESs)frombegin node *Novel* (NO) and tofinish node *Autobiography* (*AU) in Buybooks.com*, as shown in Figure 1. Here, three user traversals fromNOto AU*:* U_a. who traversed the node sequence *(NO, HI, SC, MY, IN, EL, AU)* andusers U_b, and U_c, both of whom traversed the sequence *(NO, SO, SC, MY, IN, ED, AU)* .

Figure 1: In this example, the subpaths*(NO, HI, SC)*and *(IN, EL, AU)* were traversed once:the subpaths*(NO, SO, SC)* and *(IN, ED, AU)* twice: and the subpath*(SC, MY, IN)*three times. If the frequency threshold is 3, then only the subpath*(SC, MY, IN)*qualifiesas a CES. Whereas if the threshold is set to 2, then the subpaths*(NO, SO, SC)* and *(IN, ED, AU)*qualify as well.

The problem can be formally stated as follows: given a website graph G and, a mechanismfor observing routing over G. in this work, algorithms to discover CESs will be designed. Givenspecific start and end nodes.Numerous approaches are available, that can be applied to solve this problem in theory.Thedata transfer to a large extent, on large enterprise web sites creates an enormous volume of navigationdata. It is impractical to make the direct application of current graph algorithm. Applying even log-linear algorithms, to the navigational data of large sitesresults in running times too large to be of practical value. Thus, the focus of this work isto describe a practical approach to discover CESs quickly and efficiently.

IV. APPROACH SUGGESTED IN THIS WORK

This work proposesanapproach to solve this problem. An alternative methodology for discovering the most popular traversed paths which is based on aggregating the web usage data and applying an approximate algorithm to this aggregated data. The proposed methodology discovers the most frequently traversed segments in an e-business site in a very short time (on the order of minutes to a few hours, depending on the size) and

with reasonable accuracy.The approach to CES discovery between two pages (i.e. nodes) on a site can be dividedinto three steps:

- obtaining and preparing site traversal information for analysis
- find the *k* most-traversed paths
- analyze those *k* paths to find CES

It is found that the second step, which is the mostdifficult to perform in a timely and accurate manner been simplified with suggested approach.As a result, the key contributionof this work will be the development of strategies that will allow the efficient and rapidcomputation of the k most traversed paths between a given start and end node on a website.

a. Obtaining and Preparing Site Traversal Information for Analysis

The basic source of input for CES analysis is a session log, which records everyrequest to the site and thus serves as a mechanism for observing navigation on the site.There is a significant problem with analyzing logs for user traversal information, specifically,the ever-expanding size of the log data, and the difficulty in analyzing suchlarge data volumes.

It can easily be seen that storing full session log data requires significant amount ofstorage space, especially if log data must be archived over time.Many sites experience a much higher user load. For example a log entry on a site consists of some 400 bytes of data, and each user click, onaverage, generates 10 log entries (since each embedded object, such as an image, requestedon the site is logged separately). Here, the size of the log file generated per day is10,000,000 x 10 x 400 x 10 bytes or 0.4 TB. Over a period of a one single year, the datavolume grows to almost threedigit TB. To perform meaningful analysis on this dataset, however, it is imperative that log data be accumulated over extended periods of time. Clearly, storingand querying such large data volumes is prohibitively expensive and developing algorithmsthat can operate in a practical manner on such very large data sets is a challenging issue.

Here, log data must be processed in order to separate user sessions from one another. Itis, however, well-documented in the literature [10, 4,26] that identifying user traversalsin a session log is difficult. This is primarily due to two characteristics of log files. First,the log entries corresponding to different user sessions are interleaved. Second, log entriesare not necessarily marked with a unique identifier for each session. Typically IP addressesare used to identify sessions; however, the occurrence of proxies amid, end users and websites can mask multiple users' interactions with a web site with a single IP address.

Due to the prohibitively large storage requirements and time constraints, it is impractical,indeed even impossible, to analyze the full log data to exactly identify the most popularpaths on a site. To address this issue of input data size, a smaller, graphical representationmotivates the need for the development of approximate algorithms. As we know, with the exhaustive and sequence-mining approaches, the large data volumes make this the most difficult part of the CES discovery problem to perform.In this regard following noteworthy properties need attention of researchers:

i. Small Size Requirement

The suggested approach takes as input, a *smaller,aggregated* representation of user traversal information, which greatly reduces storage requirements.For example, in one of our live-data tests, aggregation reduced the storagerequirement for a set of traversal data from approximately order of terabytes to 24 MB.While the commonly held notion that storage is cheap is indeed valid, reducing storage requirements offers a number of important benefits.

First, reducing overall storage requirements reduces storage maintenance costs. Thepurchase costs for storage run at about $ 1 per megabyte whereas the maintenance cost of each megabyte of data stored is approximately $6 per *year* [27]. The annual cost of storing it grows along with increase involume of data.

Second, storage size reduction also minimizes physical space requirements. This is animportant consideration for sites, especially in the context of the current trend to outsourcehosting (to infrastructure providers). In this scenario, the cost ofhostinga site is often tied directly to the physical space it occupies at the hosting facility, i.e. hosting a site that requires two racks to accommodate its data needs will cost significantlymore than a site requiring only one rack. As more and more E-Commerce sites outsourcethe hosting of their sites, optimizing rack space has become a serious concern and one of the most significant consumers of rack space real estate is data storage.

Third, and most importantly, the reduced input size makes analysis of the data practical.
Unaggregated traversal data simply grows too large to handle, even with a low-order polynomial algorithm .This approach tackles this aggregation of data in mind.

ii. Low Time Complexity

Suggested approach is also practical, with low-time complexity. This methodoriginate from Dijkstra's Single-Source Shortest-Path algorithm [7], without significant increases in time complexity: the average running-time is still log-linear. Since this type of analysis can be executed in batch mode off-line, loglinear complexity is clearly practical. In real terms, the running time ofthe longest-runningtest required, only 15 minutes to produce results for 100000 users sessions. Interestingly, asimilar test on the unaggregated data for the 10000 user sessions using an optimal (exact)method consumed 18 hours.

iii. Accurate, Even Though Approximate

It can easily be inferred, basedon the discussion in Property 1 , that the suggested approaches are by necessity, approximate – generatingexact results requires storing and analyzing all accumulated traversal information,based on the example , is impractical. Still, the suggested approach identifyCESs with high accuracy, quantified using a set of metrics adapted from the well-knownInformation Retrieval notions of precision and recall.

b. Finding the Critical Edge Sequences

The final task, that of analyzing the output of a k-most-traversed-path-finding algorithmto find CESs is much more straightforward than the second task.CESs can be derived from subsequences of the *k* paths in many different ways. Thesimplest way to do this is to extract all common subsequences across the *k* paths. Typically,not all of the CESs are of interest: rather they are filtered based on some applicationdependentcriteria. For example, some applications may be interested in subsequences of aparticular length, while others may impose a minimum probability threshold or frequencyof occurrence in the *k* paths. Given a set of *k* most-traversed paths, the application of thesecriteria is not difficult. In fact, it turns out that we can apply a wide variety of patternmatchingalgorithms, to this part of CESanalysis.

V. CONCLUSION

In summary, an approach which can lead to space-efficient, time-efficient, and accurate methods for finding CESs on a site.The suggested method take as input a smaller, aggregated representation ofuser traversal data compared to standard Web logs. The method has been tested on bothsynthetic as well as extensive real data. It provide a vast reduction in the storage size required and time consumed, but with minimum impact on the accuracy of the results.

The work makes an effort towards contributions to the development of cost effective, scalable e-business infrastructures that can support the future growth of the Internet. More specifically, this work contributes to one of the important aspects of e-business infrastructure development i. e. mining valuable web usage behavior,of e-business infrastructures. Contribution in this area is thedevelopment of a methodology that enables the findingof the mostly popular traversed paths in E-Commerce sites. Such a discovery will lead to the identification of critical traversal patterns at a site. This methodology enables pattern discovery on large volumes of data (e.g., terabyte range) within reasonable time frames (e.g. on the order of minutes to a few hours). Suggested methodology is both space-efficient and practical, and provides a tunable parameter, which provides an option to trade increased time and space costs for greater accuracy of results.

REFERENCES

[1]. DAIKOKU, G.SHU,L.,CRONIN, E,GARTZEN, P,.LESKELA, L.. and SIDDALL. D.. "The economic downture is not an excuse to retrench b2b efforts. " Gartner Group: http://www.gartner.com, Martch 2010.

[2]. MURPHY, K., 'stickiness' is the new gotta-have.'' http://www.internetworld.com/print/1999/03/29/news/19990329-stickiness.html.March 1999.

[3]. [3] W3C, "Extended log file format.'' http://www.w3.org/TR/WD-logfile.html,2012.

[4]. BUCHNER, A. BAUMGARTEN, M., ANAND. S. MULVENNA. M.. and HUGHES. J., "Navigation pattern discovery from internet data," in Proceedings of WE-BKDD'2014:

[5]. BORGES, J. and LEVENE, M., "Data mining of user navigation pattern," in Proceedings of WEBKDD'99: Workshop on web Usage Analysis and User Profiling, 1999.http://www.acm.org/sigs/sigkdd/processdings/webkdd99/toconline.htm.

[6]. COOLEY, R. MOBASHER, B., and SRIVASTAVA, J. "Data preparation for mining world wide web browsing patterns," Knowledge and information system, vol. 1, no.1,app.5-32,1999.

[7]. CORMEN, T., LIESERSON, C., and RIVEST, R., Introuducetion of Algorithms. McGraw Hill.2012.

[8]. KNUTH. D. The Art of Computer Programming: Fundamental Algorithms. Addison Wesley, 1997.

[9]. CHEN, M.S. Park, J..and YU,P., "Data mining for path traversal patterns in a web environment," in Proceedings of the International Conference on Distributed Computing Systems,pp. 385-392,1996.

[10]. MOSBASHER, B. JAIN, N. HAN, E..and SRIVASTAVA, J., "Web mining: Pattern discovery from world wide web transactions." Tech. Rep. 96-050.University of Minnesota, Dept. of Computer Science, Minneapolis, 1996.

[11]. SPILIOPOULU, M.. FAULSTICH, L..and WINKLER. K. "A data miner analyzing the navigational behavior of web users." In International Conference of ACAI '99: Workshop on Machine Learning in User Modelling,2013.

[12]. GAUL, C. and SCHMIDT-THIEME, L. "Mining web navigation path fragments. "in Proceedings of the 2000 WEBKDD Workshop, August 2000.

[13]. NANPOULOS, A.. KATSAROS, D., and MANOLOPOULOS, Y., "Effective prediction of web-user accesses a data mining approach." In Proceedings of the 2001 WEBKDD Workshop, August 2009.
[14]. ZAKI. M., LESH.N., and OGIHARA. M.. "Planmine: Sequence mining for plan failures," in Proceedings of the 4th Intl. Conference on knowledge Discovery and Data Mining,pp. 369-373,1998.
[15]. AGRAWAL, R. IMIELINSKI, T. and SWAMI. A., "Mining association rules between sets of iteams in large databases," in Proceedings of the 1999 ACM SIGMOD international Conference on Management of Data, app.207-216, May 1999.
[16]. AGRAWAL, R. and SRIKANT, R. "Fast algorithms for mining association rules in large databases." In proceedings of the Twentieth International Conference on very large databases (VLDT 1994), app.487-499,1994.
[17]. BAYARDO. R. and AGRAWAL. R.. "Mining the most interesting rules." in Proceedings of the Fifth ACM SIGKDD International Conference on knowledge Discovery and Data Mining,app. 145-154,1999.
[18]. KORN. F. LABRINIDIS, A. KOTIDIS. Y. and FALOUTSOS.C.. "Quantifiable data mining using ratio rules," VLDB journal,app. 254-266. 2000.
[19]. AGRAWAL, R. and SHRIKANT, R. "Mining sequential patterns," in Proceeding of the Eleventh International Conference on Data Engineering, app.3-14, March 1995.
[20]. SRIKANT. R. and ADRAWAL. R. "Mining sequential patterns: Generalization and performance improvements," in Advances in Database Technology – EDBT '96, 5th International Conference on Extending Database Technology,pp. 3-17, March 1996.
[21]. SPILIOPOULOU. M. and FAULSTICH. L. C.. "WUM: A web utilization miner. "in The World Wide Web and Databases. International Workshop WebDB '98.app. 184-203. 1998.
[22]. DESHPANDE. M. and KRYPIS.G."Selective markov models for predicting web page accesses," in Proceedings of the First SIAM International Conference on Data Mining. April 2001.
[23]. PITKOW. J. and PRIOLLI, P.. "Mining longest repeating subsequences to predict world wide web surfing," in Proceedings of USITS '99: The 2nd USENIX Symposium on Internet Technologies and System,(Boulder, Colorado), October 1999.
[24]. FLORESCU, D..LEVY.A.. SUCIU. D. and YAGOUB, K., "Optimization of run-time management of data intensive web sites," in Proceedings of the 25th VLDB Conference, app. 627-638, September 1999.
[25]. DATTA, A..VANDERMEER. D. RAMAMRITHAM, K... and NAVATHE, S.."Toward a comprehensive model of the content and structure of and user interaction over a web site," in Proceedings of the VLDB Workshop on Technologies for E-Services. (Cairo, Egypt). September 2011.
[26]. SHSHABI, C.. ZARKESH, A.,ADIBI, J., and SHAH, V., "Knowledge discovery from users web-page navigation," in Proceedings of RIDE'97-Seventh International Workshop on Research Issues in Data Engineering,1997.
[27]. SIMPSOM. D., "Corral Your Storage management costs."Datamation, pp. 88-93, April 1997.