Mining Indirect Associations in Web Data*

Pang-Ning Tan
Department of Computer Science
University of Minnesota
Minneapolis, MN 55455
ptan@cs.umn.edu

Vipin Kumar
Department of Computer Science
University of Minnesota
Minneapolis, MN 55455
kumar@cs.umn.edu

ABSTRACT

Analysis of association is an important Web mining technique because it can provide useful insight into the navigational behavior of Web users. E-tailers can use this information to develop strategic marketing plans and to re-structure their Web site in order to enhance the browsing experience of their customers . Previous work on mining Web associations has focused primarily on finding frequent access patterns in the data. These patterns can be generated by Web users who share similar information goals or by those with varying interests. Since Web association patterns consider only co-occurrences in data, it is difficult to identify patterns generated by one group of Web users but not by the others. Another drawback of the existing approach is that it does not adequately address the impact of Web site structure on the support of a Web page. As a result, the majority of Web association patterns discovered using conventional techniques contain the home page or other reference pages that have multiple outgoing links. In this study, we apply a new mining technique called indirect association to Web usage data. This novel technique is capable of combining the various association patterns into a more compact structure. It can also capture both positive and negative correlations that exist in the data. We demonstrate the applicability of this technique on Web data from both commercial and research institutions. Our analysis shows very promising results, especially in terms of identifying Web users with distinct interests.

1. INTRODUCTION

The unprecedented growth of the World Wide Web has revolutionized the way most commercial organizations conduct their businesses today. Nowadays, it is becoming increasingly common that the first point of interaction between a customer and an organization is at a Web site. As the number of online Web traffic grows, so does the volumes of data collected at the Web servers. This has fueled a tremendous amount of interest in applying data mining techniques to discover hidden patterns in the Web clickstream data.

Association rules [1] and sequential patterns [3] are two notable types of Web patterns [9, 21] that can bring added values to an e-commerce organization. These *Web association patterns* can reveal valuable information about the navigational behavior of users who are accessing their Web sites. Previous research work in mining association patterns has focused primarily on discovering patterns that occur frequently in the data, i.e. patterns with high *support*[1]. Any patterns that do not have sufficient support are assumed to be statistically insignificant and therefore, eliminated. However, there are situations in which such a filtering strategy may end up removing a huge amount of useful information. For example, patterns involving anti-correlated sets of pages can be quite informative even though they seldom co-occur together. These anti-correlated patterns may represent the navigational behavior of distinct groups of Web users.

A second, but equally significant, problem with the current association mining technique is that it does not adequately address the impact of Web site structure on the support of a pattern. For instance, Web pages that are located close to the home page will more likely have higher hit rates compared to those that are located further away. In addition, Web pages that have many outgoing links (hub pages) also tend to have higher support. As a result, the bulk of the Web association patterns discovered using conventional techniques consist of such reference pages (i.e., the home page and hub pages). So far, it is still unclear as to how to deal with patterns containing these pages. One trivial solution is to remove the reference pages during preprocessing. This is not a viable solution primarily because some of the reference pages can be quite informative. Alternatively, one can use a lower support threshold to ensure that most of the interesting non-reference pages are captured by the Web patterns. The drawback here is that a large number of uninteresting patterns due to co-occurrences among reference

*This work was partially supported by NSF grant # ACI−9982274 and by Army High Performance Computing Research Center contract number DAAH04−95−C−0008. The content of this work does not necessarily reflect the position or policy of the government and no official endorsement should be inferred. Access to computing facilities was provided by AHPCRC and the Minnesota Supercomputing Institute.

[1]The support of a pattern is the fraction of data set for which the pattern is observed.

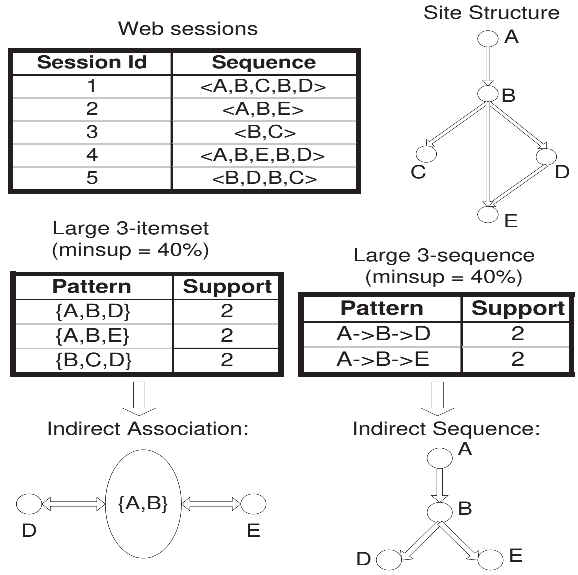

Figure 1: Example of Sequential and Non-Sequential Indirect Associations.

pages will be generated. One must then filter out the uninteresting patterns either objectively, with an appropriate interest measure [22], or subjectively, with domain knowledge [18, 8]. Instead of filtering, another possible solution is to group together patterns that have a common set of reference pages into higher order structures. These structures would contain richer amount of information and thus, allow an analyst to have a better understanding of the derived patterns.

In this paper, we have applied a new data mining technique called indirect association to Web sequence data. This technique was originally introduced in [24] to capture higher order dependencies in market-basket type of transactions. Indirect association combines patterns generated by conventional association pattern discovery algorithms into higher order structures, and is capable of capturing both positive and negative dependencies in data. The main contribution of our current work is to extend this idea to sequential data. In particular, from the Web Mining perspective, sequential indirect associations allow us to characterize the distinctive groups of Web users who share similar traversal subpaths. They can also be used to detect interesting bifurcation (splitting) points in the Web site structure, i.e., Web pages that serve as the divergence of various user interests. Knowledge about the different groups of Web users and pages that fork out the varying interest can help analysts to take the appropriate actions (for target marketing, placement of banner advertisement, etc.) that will benefit the e-commerce organization as well as assisting Web site administrators in re-organizing the structure of their Web site to enhance the browsing experience of their site visitors. We illustrate the idea of indirect association in the example below.

Example 1. Consider the table of Web sessions and site structure given in Figure 1. First, we find all the sequential and non-sequential patterns that occur frequently in the Web data. By imposing a minimum support threshold equals to 40%, any patterns that appear in less than 2 sessions can be discarded. The sequential and non-sequential frequent patterns of size 3 are given in Figure 1. For the non-sequential case, we would discover that the page D co-occurs frequently with pages A and B. Similarly, E also appears frequently with both A and B. Without any prior knowledge about the structure and content of the pages, we would expect D and E to appear together quite often. However, since D and E are infrequent (and negatively correlated), we say that they are indirectly associated via the mediator set $\{A, B\}$. We can also find an indirect association between A and C via $\{B, D\}$. For the sequential case, we first discover a pair of frequent traversal paths, $(A \rightarrow B \rightarrow D)$ and $(A \rightarrow B \rightarrow E)$. Since D and E do not appear together often (in any order), we declare them to be indirectly associated via the sequence $(A \rightarrow B)$. In this example, B becomes the bifurcation point for D and E. Notice that any sequential indirect association has a corresponding non-sequential indirect pattern, but not vice-versa. For example, there is no sequential indirect association between A and C via $B \rightarrow D$.

This paper is organized in the following way. In section 2, we introduce the formal definition of indirect association for sequential and non-sequential data. Next, we present a level-wise algorithm for mining these patterns. We also describe how demographic data can be used to enrich the information conveyed by indirect associations. We then demonstrate the applicability of this technique on real-world data sets in Section 4. Finally, we conclude with a summary of our results and directions for future research.

2. BACKGROUND

2.1 Definition

Let $I = \{i_1, i_2, \cdots, i_d\}$ denotes a set of literals called items (or events, for sequential data) and C be a non-empty subset of these items called an itemset. An itemset containing k items is also known as a k-itemset. Also, suppose T is the set of all transactions, where each transaction $t \in T$ is a subset of I. The support of an itemset C, $sup(C)$, is defined to be the fraction of all transactions T that contain C. An itemset is large (frequent) if its support is greater than a user-specified threshold t_f.

A sequence is an ordered list of itemsets, $s = < s_1, s_2, \cdots, s_n >$, where each itemset s_j is an element of the sequence. The length of the sequence s is $|s| = n$ while the size of an element s_j ($|s_j|$) corresponds to the number of items (events) contained in the element. We will use the terms item and event interchangeably throughout this paper. A sequence is said to be non-empty if it contains at least one element (i.e. $|s| > 0$). Note that an item can appear only once in an element, but can occur multiple times in different elements of a sequence. Items in an element are also assumed to be sorted in lexicographic order. An item x_i that appears only once throughout a sequence s is called a *non-repeating* item.

A sequence $t = < t_1, t_2, \cdots, t_m >$ is called a subsequence of s if there exist integers $1 \leq j_1 < j_2 < \cdots < j_m \leq n$ such that $t_1 \subseteq s_{j_1}, t_2 \subseteq s_{j_2}, \cdots, t_m \subseteq s_{j_m}$. A sequence database D is a set of tuples $< sid, t >$ where sid is the sequence identifier and t is a sequence. A tuple $< sid, t >$ is said to contain a sequence s if s is a subsequence of t. The support of a sequence s, $sup(s)$, is defined as the fraction of all tuples in D that contain s.

A sequence with k items, where $k = \sum_j |s_j|$, is called a k-sequence. The concatenation of two sequences s and t, denoted as st, is a sequence of length $|s| + |t|$ and consists of all elements of s immediately followed by the elements of t. A sequence w is a prefix sequence of s, denoted $w \sqsubset s$, if there exists a non-empty sequence y such that $s = wy$. w is a minimal prefix sequence of s if $w \sqsubset s$ and $|w| = 1$. Conversely, w is a maximal prefix sequence of s if $w \sqsubset s$ and $|w| = n-1$, where $|s| = n$. Similarly, y is a suffix sequence of s, denoted as $y \sqsupset s$, if there exists a non-empty sequence w, such that $s = wy$. It follows that the minimal and maximal suffix sequences of s correspond to the suffix sequences $y \sqsupset s$ such that $|y| = 1$ and $|y| = n - 1$ respectively.

An item x is a prefix item of sequence $< a_1, a_2, \cdots, a_n >$ if $x \in a_1$ and $|a_1| = 1$. On the other hand, x is said to be a suffix item of the sequence if $x \in a_n$ and $|a_n| = 1$. In general, x is called an *end item* of the sequence s if it is either a prefix or suffix item of s.

2.2 Indirect Association for Itemsets

Definition 1. A pair of items, a and b, is said to be indirectly associated via a mediator set M if the following conditions hold :

1. $sup(\{a, b\}) < t_s$ (Itempair Support condition)
2. There exists a non-empty set M such that :
   (a) $sup(\{a\} \cup M) \geq t_f, sup(\{b\} \cup M) \geq t_f$ (Mediator Support condition).
   (b) $d(\{a\}, M) \geq t_d,\ d(\{b\}, M) \geq t_d$ where $d(P, Q)$ is a measure of the dependence between the itemsets P and Q (Mediator Dependence condition).

Condition 1 is needed because an indirect association is significant only if both items rarely occur together in the same transaction. Otherwise, it makes more sense to characterize the pair in terms of their direct association. Alternatively, condition 1 can be modified to test for independence between items a and b. However, it is often the case that itempairs that have very low support values are either independent or negatively correlated with each other. Thus, condition 1 is sufficient to effectively discover indirect relationship between independent or negatively correlated itempairs.

Condition 2(a) can be used to ensure that the mediator set contains items that are statistically significant. Support also has a nice downward closure property that allows us to reduce the exponential number of candidate mediators.

Condition 2(b) ensures that only itemsets that are highly dependent on both a and b are used to form the mediator set M. For instance, suppose there exists an item k that appears in every transaction (e.g. the home page). Without the mediator dependence condition, any itempairs that are infrequent will be indirectly associated via k. Thus, condition 2(b) is necessary to prevent the generation of spurious indirect associations via uninteresting mediators. Currently, there are many interest measures that can be used to represent the degree of dependencies among attributes of a dataset. One such measure is Pearson's linear correlation coefficient, ϕ. For binary pairs of attributes, it can be shown that within certain range of support values [2], the correlation between the pair X and Y, $\phi_{X,Y}$, can be expressed in terms of an interest factor, $I(X, Y) \equiv \frac{P(X,Y)}{P(X)P(Y)}$ [6, 5], and the joint probability (support) of the pair, i.e. :

$$\phi_{X,Y} \approx \sqrt{\mathrm{I(X, Y) \times P(X, Y)}} \equiv IS(X, Y)$$

We will use the right-hand side of the above expression, called the IS measure, to be the dependence measure in Condition 2(b) [22]. This measure is desirable because it takes into account both the statistical dependence and statistical significance of a pattern. We do not use $\phi_{X,Y}$ as the dependence measure because it treats both the presence and absence of items in the same manner (as shown in the example below). In many data mining applications, the presence of items are more important than their absence.

Example 2. In the table below, the correlation between X and Y are the same for both (a) and (b), even though the joint support for X and Y in (b) is higher than (a):

$$\phi_{X,Y} = \frac{0.1 \times 0.6 - 0.15 \times 0.15}{0.25 \times 0.75} = 0.1125 \qquad (1)$$

This is because the correlation measure ϕ is symmetric in terms of both presence and absence of items. In contrast, the IS measure for (b) is higher than (a):

$$IS_{(a)} = \sqrt{\frac{0.1 \times 0.1}{0.25 \times 0.25}} = 0.4 \qquad (2)$$

$$IS_{(b)} = \sqrt{\frac{0.6 \times 0.6}{0.75 \times 0.75}} = 0.8 \qquad (3)$$

(a)	$Y = 1$	$Y = 0$	(b)	$Y = 1$	$Y = 0$
$X = 1$	0.1	0.15	$X = 1$	0.6	0.15
$X = 0$	0.15	0.6	$X = 0$	0.15	0.1

Nevertheless, our indirect association formulation can accommodate other interest measures, such as Piatetsky-Shapiro's rule-interest, J-measure and Gini index, which have been shown to be equally good at capturing statistical correlation [22]. The above formulation can also be extended to the case where a and b are itemsets rather than single items.

2.3 Indirect Association for Sequences

Let a be a non-repeating, end item for the sequence $s_1 =< a_1, a_2, \cdots, a_n >$, and b is a non-repeating, end item for the sequence $s_2 =< b_1, b_2, \cdots, b_n >$. Furthermore, let a_i and b_j denote the elements of s_1 and s_2 containing the items a and b, respectively (i.e. $a_i = \{a\}$ and $b_j = \{b\}$).

Definition 2. A pair of items a and b are said to be indirectly associated via a mediator sequence w if $s_1 = a_i w$

[2]when $P(X) \ll 1$, $P(Y) \ll 1$ and $\frac{P(X,Y)}{P(X)P(Y)} \gg 1$.

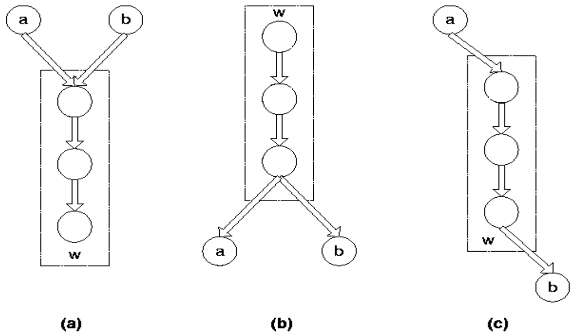

Figure 2: Types of sequential indirect asssociation between a and b via a mediating sequence w: (a) Type C (b) Type D (c) Type T.

(or $s_1 = wa_i$), $s_2 = b_j w$ (or $s_2 = wb_j$) and the following conditions are satisfied:

1. $sup(\{a, b\}) < t_s$ (Itempair Support condition).

2. $sup(s_1) \geq t_f$ and $sup(s_2) \geq t_f$ (Mediator Support condition).

3. $d(a_i, w) \geq t_d$ and $d(b_j, w) \geq t_d$ (Dependence condition).

The first condition ignores the order in which a and b appears in the data sequences. The second condition guarantees that only the commonly traversed subpaths are used in our analysis[3]. For the third condition, we will use the IS measure to define the dependencies between elements and sequences.

In this paper, we are interested in three types of sequential indirect associations (Figure 2):

1. **Type-C (Convergence)** - if a_i and b_j are the minimal prefix sequences for s_1 and s_2 respectively.

2. **Type-D (Divergence)** - if a_i and b_j are the minimal suffix sequences for s_1 and s_2 respectively.

3. **Type-T (Transitivity)** - if a_i is the minimal prefix sequence for s_1 and b_j is the minimal suffix sequences for s_2, or vice-versa.

It is straightforward to extend this formulation to indirect association between elements having more than one item.

[3]Pitkow et al [15] observed that many paths occur infrequently, often as a result of erroneous navigation. By using the commonly traversed subpaths, one can preserve the benefits of sequential nature of the paths while being robust to noise.

2.4 Related Work

It is important to point out that there are several formulation to the Web sequence mining problem. Each formulation differs in terms of how *Web transactions* can be constructed from the original session information. A naive approach simply ignores the sequential nature of the data and considers each transaction to be the set of pages accessed in the session.

A second formulation converts each session into a set of maximal forward references in order to filter out the effect of backward references [7]. For example, the first session of figure 1 will generate two maximal forward references, $< A, B, C >$ and $< A, B, D >$.

A third formulation considers each individual page access in a session as events of a Web sequence [3, 13, 19, 14, 4, 11]. Spiliopoulou et al [19] and Pei et al [14] combines the various sequences into a compact tree data structure to facilitate querying and mining of Web association patterns. Agrawal et al [3] and Mannila et al [13] relaxes the definition of an element of a sequence to a wider time window. Garofalakis et al.[11] uses regular expressions to specify constraints on items (events) that may appear as elements of a frequent sequence. The algorithms proposed in [3, 13, 11] are based on the generate-and-count paradigm, i.e. candidate patterns are initially generated prior to actual support counting.

A fourth formulation considers each session as an alternating series of vertices (pages) and edges (hyperlinks)[17]. This approach assumes that knowing the sequence alone is insufficient because there could be more than one link connecting the same pair of pages and each link may contain different information.

The notion of indirect association proposed in this paper is equally applicable to any one of the transaction formulation above. For brevity, we describe indirect association only in the context of the first and third formulation.

Indirect association is closely related to the concept of negative association rules [16]. In both cases, we are dealing with itemsets that do not have sufficiently high support. A negative association rule discovers what are the set of items a customer will not likely buy, given that he/she has bought a certain set of other items. Typically, the number of negative association rules in a data set can be prohibitively large, and the majority of them are uninteresting to the data analyst. In [16], Savasere et al. has proposed the use of item taxonomy to decide what constitutes an interesting negative association rule. Their intuition was that items belonging to the same product family are expected to have similar types of associations with other items. Thus, if the observed support of a pattern is significantly smaller than its expected support (computed based on the item taxonomy), they conclude that an interesting negative association rule exists among the items. In contrast, our approach assumes that an interesting negative association exists if the items share a common set of other items, which may or may not belong to the same product family.

The idea of indirect association presented in this paper provides a methodology to reduce the large number of discovered patterns by grouping them according to items (events) they have in common. Even though the idea of grouping association patterns is not new, our work differs from others in terms of the types of patterns being grouped and how the summarized patterns are represented. In [25], Toivonen et al. developed the notion of a rule cover, which is a small set of association rules that covers the entire database. Clustering was also used to group together similar rule covers. In [12], Liu et al. have used the χ^2 test to reduce the number of association rules, and direction setting rules to summarize the remaining association rules. Both approaches are different from our work because they were interested in grouping together association rules that have the same rule consequent, whereas indirect association are used to summarize frequent itemsets and sequential patterns.

3. ALGORITHM

An algorithm for mining (non-sequential) indirect association between pairs of items is shown in Table 1. There are two phases in this algorithm. During the first phase, all large itemsets are initially derived using standard frequent itemset generation algorithm such as Apriori [2]. The large itemsets L_k are then used to generate candidate indirect associations for pass $k+1$, i.e. C_{k+1}. Each candidate in C_{k+1} is a triplet, $< a, b, M >$, where a and b are the indirect pairs, associated via the mediator M. During the join step, a pair of large k-itemsets, $\{a_1, a_2, \cdots, a_k\}$ and $\{b_1, b_2, \cdots, b_k\}$, can produce a candidate indirect association $< a, b, M >$ if the two itemsets have exactly $k-1$ items in common. Since the itemsets are created by joining two large itemsets, the mediator support condition is trivially satisfied. The remaining steps (5 through 7) are used to validate the two additional conditions for indirect association. If $|L_1| = N$ is not too large, we may initially create an $N \times N$ support matrix for all pairs of frequent items, (a, b). Thus, the itempair support condition in step 6 can be easily verified by looking up the content of the corresponding matrix entry. Alternatively, we may store the candidate indirect itempairs in a hash tree and performs an additional pass over the data set in order to count the actual support of each candidate itempair. The latter approach can be used to handle the case where a and b are itemsets rather than single items.

We now briefly describe the complexity of the algorithm. The candidate generation step can be quite expensive, because it requires at most $O(\sum_k |L_k| \times |L_k|)$ join operations. (Note that the join operation in Apriori is less expensive because it combines only itemsets that have the same $k-1$ prefix items. We do not have the luxury to do this because the indirect item can appear anywhere in a frequent itemset.) However, we have implemented various techniques to reduce the number of join operations. For example, given a frequent itemset $\{a_1, a_2, \cdots, a_k\}$, we only need to perform the join operation on the rest of the frequent itemsets $\{b_1, b_2, \cdots, b_k\}$ for which the condition $a_2 \le b_1$ is satisfied. This condition is applicable because all items in an itemset are sorted according to lexicographic order. The candidate count steps (steps 5 through 7) are less expensive if the $N \times N$ itempair support matrix is available. Otherwise, the complexity of this step can be as expensive as the candidate counting step of the Apriori algorithm.

After generating the indirect associations, a post-processing step can be used to improve our understanding of the patterns. First, we may add user demographic information to the patterns (or to the initial data set) to explicitly find indirect association among distinctive groups of users. This user demographic information may come from data sources other than the original Web server logs. Let V be the set of demographic attributes. For each indirect association (a, b, M), we look for attribute sets $Q \subset V$ and $R \subset V$ such that $\{a\} \cup M \cup Q$ and $\{b\} \cup M \cup R$ are also frequent. This allows us to identify the two distinct groups of users who are indirectly associated via M. Another approach we have taken is to combine the various indirect associations that share a common mediator. For example, suppose (a, b) and (a, c) are indirectly associated via the mediator M. We can construct a graph $G = (V, E)$ where $V = \{a, b, c\}$ and E is the edges between the vertices in V. Each edge $e = (v_i, v_j) \in E$ indicates whether a direct or indirect association exists between the two connected vertices. This association graph is a compact representation of dependence relationships among all items (itemsets) that share a common mediator. We have implemented an indirect association viewer to visualize the graphs for each mediator. Such a visualization tool allows an analyst to have a better understanding of the derived patterns. Note that unlike other association pattern viewers [26], our technique considers both the support and dependencies among various itemsets. Furthermore, our viewer is mediator-centric, instead of building a global graph of all items.

Our algorithm for mining sequential indirect association is similar to the one given in Table 1. In this case, the L_k's correspond to frequent sequences generated using a sequential pattern algorithm such as GSP [20]. During the join step, a pair of sequences s_1 and s_2 creates an indirect association (a, b, w) only if a and b are non-repeating, end items for s_1 and s_2 respectively. This follows directly from our definition of sequential indirect association. Since we are only interested in indirect relationship between end items, we can restrict the join operation to sequences of length ≥ 2 and between sequences that have the same number of items. For example, the sequence $(A) \to (C) \to (DE)$ will join with another sequence $(C) \to (DE) \to (F)$ to create a candidate indirect pair of type T between A and F via the mediating sequence $(C) \to (DE)$. However, we will not combine the first sequence above with $(FC) \to (DE)$ nor $(FG) \to (C) \to (DE)$.

4. EXPERIMENTAL RESULTS

To demonstrate the utility of indirect associations, we have performed two sets of experiments on the Web server logs from the University of Minnesota Computer Science department and from an e-commerce organization. Table 2 shows a summary of the data set description and the threshold parameters chosen for our experiments.

First, we illustrate the relationship between statistical correlation ϕ and IS measure in Figures 3 and 4. In both of these figures, we plot IS against ϕ using pairs of pages (p_i, p_j) randomly selected from all the Web pages at the Web site. We have restricted the samples to those pairs that have support count of at least 1 sequence. The overall size of the samples is 100,000 pairs. Our results show that when ϕ is greater than 0.2, both measures are linearly correlated with each

Table 1: Algorithm for mining (non-sequential) indirect association between pairs of items.

```
Extract the large itemsets, L_1, L_2, ··· L_n, using standard mining algorithms.
P = ∅
for k = 2 to n do
    C_{k+1} ← join(L_k, L_k)
    for each (a, b, M) ∈ C_{k+1} do
        if (sup({a, b}) < t_s and d({a}, M) ≥ t_d and d({b}, M) ≥ t_d
                P = P ∪ (a, b, M)
    end
end
```

Table 2: Data set summary and threshold parameters for indirect association algorithm.

Dataset	t_s	t_f	t_d	# items	# sequences
CS department	0.01	0.1	0.2	91443	34526
E-Commerce	0.005	0.05	0.2	6664	143604

other. In fact, the correlation between the two measures are very close to 1.

4.1 Non-sequential Indirect Association

For this experiment, we used the University of Minnesota Computer Science department Web server logs. The server logs are initially preprocessed to identify Web sessions. Noise due to Web robot accesses are removed using the Web robot prediction models described in [23]. The sessions are then converted into market-basket type transactions by ignoring the sequential order in which the Web pages appear in the sequence. After applying the indirect association algorithm, the derived patterns are visualized using our indirect association viewer.

We observe several interesting groups of related Web pages that share the same mediator set. These groups reflect the navigational patterns of Web users with distinct information needs. Figure 5 illustrates one such example. The mediator set (i.e. the vertex at the center of the graph) contains the Computer Science department homepage and the graduate student information page. Notice that we can divide the pages that commonly co-occur with this mediator into several distinct groups. The pages `/grad-info/recletter.pdf`, `/grad-info/finapp.pdf` and `/grad-info/survey-res-inf.pdf` are application materials for the Computer Science graduate degree program. These pages have very high support with each other. Users who access them are most likely prospective graduate students who are interested in applying for the graduate school program in this department. `/csgsa` is the home page for the Computer Science Graduate Student Association. The Web page `/grad-info/wpe-memo-2000.pdf` would be of interest to current students who are planning to take their PhD preliminary written examination while `/contact-info` contains the contact information for the Computer Science department. The last group of pages, `/Research` and `/reg-info/csMinor.html`, may represent users with varying interest. We have found many other tight clusters of pages using this mediator-centric approach. Other examples of the discovered indirect associations are shown in Figures 6 and 7.

We are also interested in finding indirect association among users with different demographic features. To do this, we have identified several demographic features that may potentially characterize the various groups of users traversing the Web site. These features include the hostname of the client (e.g. UMN, GOV or AOL), the type of browser used (e.g. Netscape or Internet Explorer) and the referrer field (which was used to indicate how the user arrives at the Web site, e.g via a search engine or other external Web sites). We have applied the frequent itemset generation algorithm on the data set that contains both session and demographic features (using a lower minimum support threshold). We use the previously found indirect associations and examine their demographic decomposition. For example, given that (a, b, M) is an indirect association, we look for demographic features X and Y (where $X \cap Y = \emptyset$) such that both $\{a\} \cup X \cup M$ and $\{b\} \cup Y \cup M$ are frequent itemsets. However, we do not discover any significantly interesting patterns with this approach. This is more likely due to our limited choice of demographic features, which may not correspond to the characterizing features of the observed user groups. We hope to expand the set of demographic features of the Web users in our future work.

4.2 Sequential Indirect Association

This section describes the results of applying our sequential indirect association technique to Web sequence data from an e-commerce organization (see Table 3). Here, we do not encounter the session identification problem as before because session identifiers are embedded in the URLs of the dynamically-generated HTML pages. We found that almost all of the Web sequences are of type D (Divergence), indicating a significant divergence in the Web traffic at these pages due to varying user interests [4]. For example, the `Auto` page is the bifurcation point for Web users who are interested in `Stereos` or `Radar Detectors`. This suggests that Web users who visits the `Stereos` page are different from those who visits the `Radar Detectors` page. Note that the `Auto` page has direct hyperlinks to both `Stereos` and `Radar Detectors` Web pages as shown in Figure 8. Since they are both *sibling* pages, it is somewhat surprising to find that

[4] The support values are omitted here to protect the information integrity of the e-commerce organization.

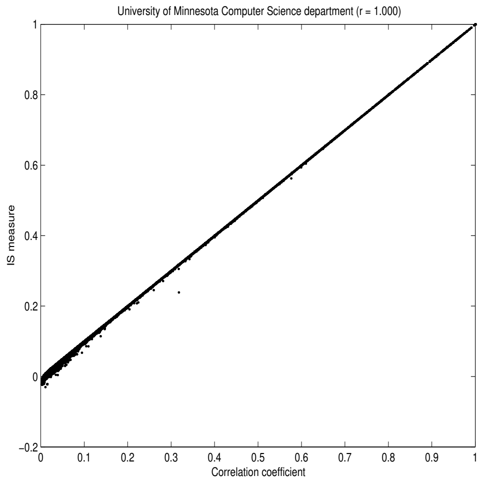

Figure 3: Comparison between IS-measure and Pearson's correlation coefficient (for CS deparment web logs).

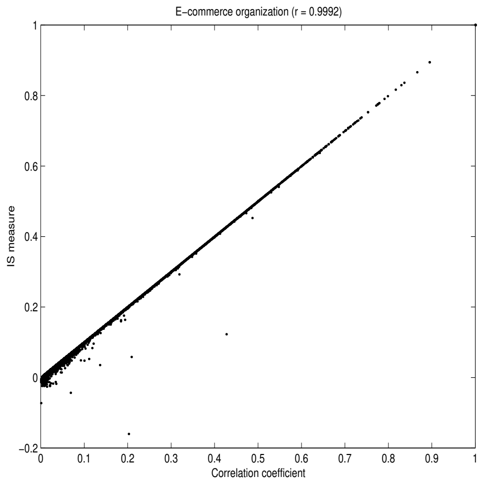

Figure 4: Comparison between IS-measure and Pearson's correlation coefficient (for e-commerce web logs).

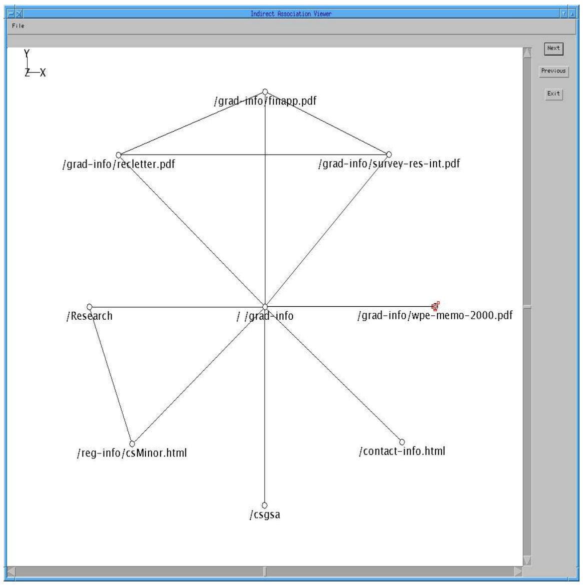

Figure 5: Indirect association among visitors to the graduate student information Web page.

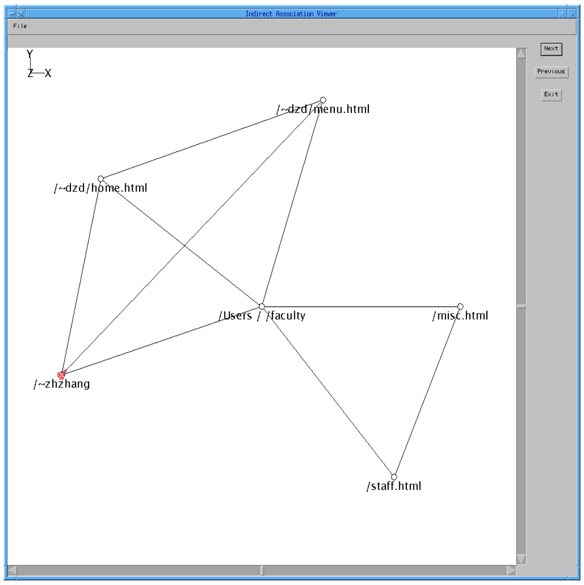

Figure 6: Indirect association among visitors to the faculty Web page.

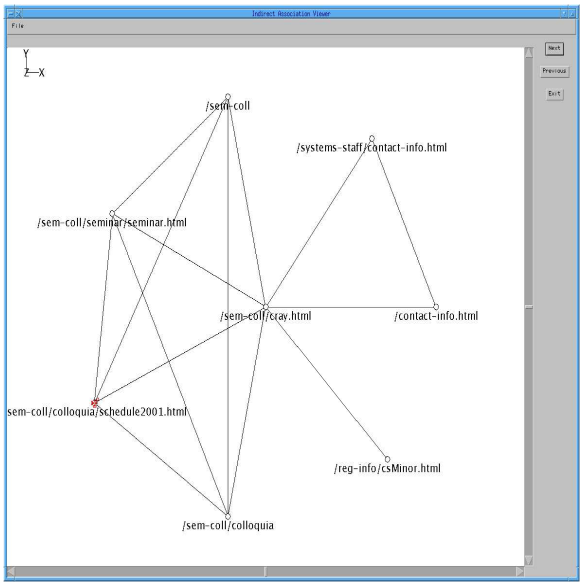

Figure 7: Indirect association among visitors to the seminar/colloquium Web page.

both pages are rarely visited together.

Another interesting sequential indirect association is between the `Hunting` accessories Web page and the `Aerobics` Web page via the homepage and the `Sporting Goods` page. Unlike the previous example, these two pages are not directly linked to their bifurcation page. In fact, their parent nodes (i.e., the `Recreation` and `Exercise` pages) are not indirectly associated with each other because their support exceeds the minimum itempair support threshold. This suggests that visitors to the `Hunting` page could still be interested in the `Exercise` page, but they are less likely interested in the `Aerobics` page. Such information can potentially influence the type of action analysts should take on the various groups of Web users - actions such as where to strategically place a banner advertisement to target specific groups of Web users. Placing it too close to the home page could be costly and ignored by most Web users. Conversely, if it is placed too far from the home page, some of the targeted Web users could miss it.

4.3 Performance Evaluation and Threshold Selection

Our experiments were performed on a 700MHz Pentium III machine with 4 GB of RAM. The computation time for the two experiments above using the threshold parameters given in Table 2 are 41.62 seconds (for non-sequential indirect association) and 3201.2 seconds (for the sequential indirect association) respectively. Note that these computation times exclude the time for generating the frequent itemsets (using the Apriori algorithm) and frequent sequences (using the GSP algorithm). Furthermore, we have observed that for both data sets, if no indirect associations are produced during pass k, then there will not be any new indirect associations generated for pass $k+1$ and higher (Figure 9). Therefore, we can terminate the algorithm at pass k.

Another important observation is that almost all of the indirectly associated itempairs are generated when combining itemsets (or sequences) of size 2 or 3 as shown in Figure 10. Larger-sized itemsets (and sequences) tend to produce larger-sized mediators for indirect itempairs that have already been found during the first two passes. As a result, it is often sufficient to generate indirect associations using frequent itemsets (or sequences) up to size 4 or 5. The runtime of the algorithm depends primarily on the choice of t_f (i.e., the minimum support threshold), and is unaffected by both t_d (i.e., mediator dependence threshold) and t_s (i.e., itempair support threshold).

To give an idea of how sensitive are the indirect associations to the choice of threshold parameters, we repeated our experiments on the non-sequential data set (from the University of Minnesota Computer Science department) using various t_s and t_d thresholds. The results of these experiments are displayed in Figures 11 and 12, respectively.

We can also use the statistical χ^2 test to determine an appropriate value of t_d. For binary pairs, the χ^2 measure is closely related to statistical correlation ϕ via the following equation, $\phi = \sqrt{\chi^2/N}$. For example, the χ^2 cutoff value at 95% confidence level with one degree of freedom is 3.84. For our two datasets, with $N = 34526$ and $N = 143604$, this cutoff value corresponds to ϕ values of 0.0207 and 0.0101. Using the IS-ϕ plots of Figures 3 and 4, any choice of t_d above 0.1 would pass the χ^2 test.

5. CONCLUSIONS

In this paper, we have applied a novel technique called indirect association to Web sequence data. This technique combines similar association patterns into a more compact structure and looks for independent or negatively correlated components within the structure. Our experiments using Web logs from a research institution and an e-commerce organization show very promising results. In particular, this technique can potentially distinguish the various interests of Web users who are traversing a particular Web site.

For future research, we are investigating the possibility of deriving indirect association using compact data structures such as FP-trees or aggregate trees. For instance, leaf nodes that share a common subpath to the root may form candidate mediators for the indirect association. Such techniques can potentially improve the performance of our algorithm especially when dealing with indirect association between itemsets or elements with more than one item. We are also interested in evaluating the use of bifurcation points for predictive modeling. Currently, Markov models have been used quite extensively to predict the access behavior of Web users [15, 10]. However, higher order Markov models are often needed to achieve better prediction accuracy. Bifurcation points can be potentially used to truncate some of the states in the high-order Markov models, while retaining their predictive accuracy.

6. REFERENCES

[1] R. Agrawal, T. Imielinski, and A. Swami. Database mining: a performance perspective. *IEEE Transactions on Knowledge and Data Eng.*, 5:914–925, 1993.

[2] R. Agrawal and R. Srikant. Fast algorithms for mining association rules. In *Proc. of the 20th VLDB Conference*, Santiago, Chile, 1994.

[3] R. Agrawal and R. Srikant. Mining sequential patterns. In *Proc. of Int. Conf. on Data Engineering*, Taipei, Taiwan, 1995.

[4] J. Borges and M. Levene. Mining association rules in hypertext databases. In *Proc. of the Fourth Int'l Conference on Knowledge Discovery and Data Mining*, August 1998.

[5] T. Brijs, G. Swinnen, K. Vanhoof, and G. Wets. Using association rules for product assortment decisions : A case study. In *Proc. of the Fifth Int'l Conference on Knowledge Discovery and Data Mining*, San Diego, August 1999.

[6] S. Brin, R. Motwani, and C. Silverstein. Beyond market baskets: Generalizing association rules to correlations. In *Proc. of 1997 ACM-SIGMOD Int. Conf. on Management of Data*, Tucson, AZ, 1997.

[7] M.S. Chen, J.S. Park, and P.S. Yu. Efficient data mining for path traversal patterns. *IEEE Transactions on Knowledge and Data Eng.*, 10(2):209–221, 1998.

Table 3: Examples of sequential indirect associations derived using the Web log from an e-commerce organization.

#	Type	a	w	b	$d(a, w)$	$d(b, w)$
1	D	Camcorders	Home → Electronics → Video	13" TV and VCR Combo	0.29	0.21
2	D	Camcorders	(Electronics, Video)	27" TV	0.25	0.20
3	D	Stereo	Home → Electronics → Auto	Radar Detector	0.25	0.34
4	D	Hunting	Home → Sporting Goods	Aerobics	0.23	0.21
5	D	Scanner	Home → Electronics → Computer	Multimedia Computer	0.20	0.27
6	D	Shower curtains	Domestics → Bath Shop	Towel set	0.43	0.30
7	D	Bedroom furniture frame	Home & accessories → Furniture	Oak nightstand	0.22	0.20
8	D	Speakers	Electronics → Stereo	CD boombox	0.34	0.20
9	D	Women's gown set	Home → Apparel	Men's boots	0.22	0.21
10	D	Cordless phone	Home → Telephones	Answering device	0.30	0.34

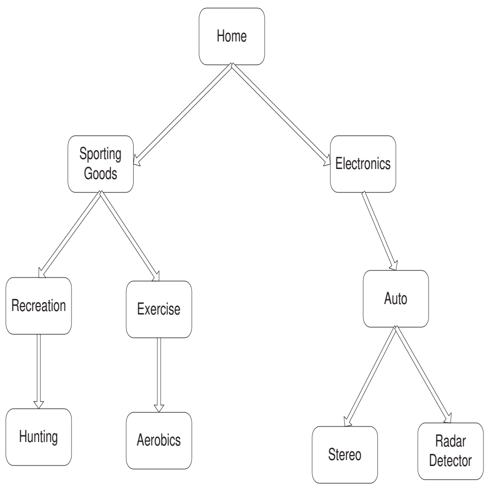

Figure 8: A subgraph of the e-commerce Web site structure.

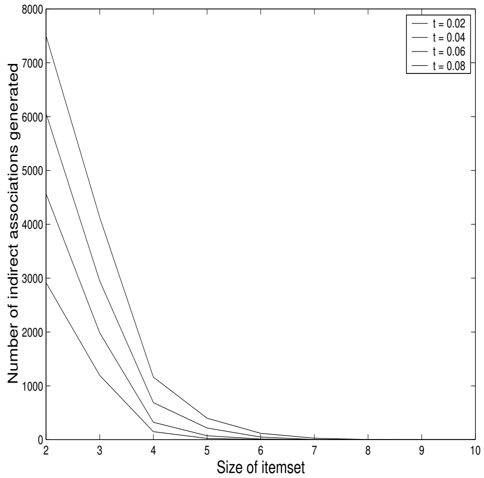

Figure 9: Total number of non-sequential indirect associations generated at various frequent itemset sizes. The threshold t refers to the minimum itempair support threshold.

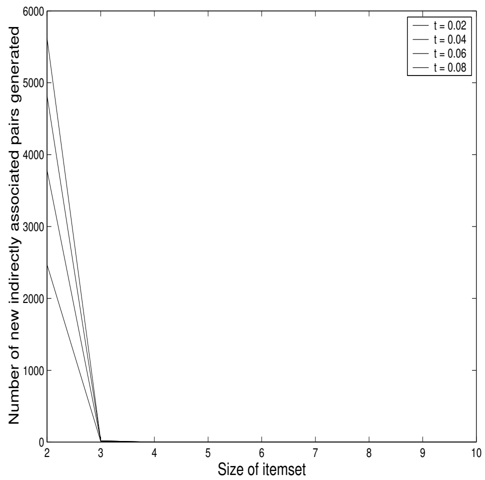

Figure 10: Total number of non-sequential indirect itempairs (irrespective of the mediator sets) generated at various frequent itemset sizes. The threshold t refers to the minimum itempair support threshold.

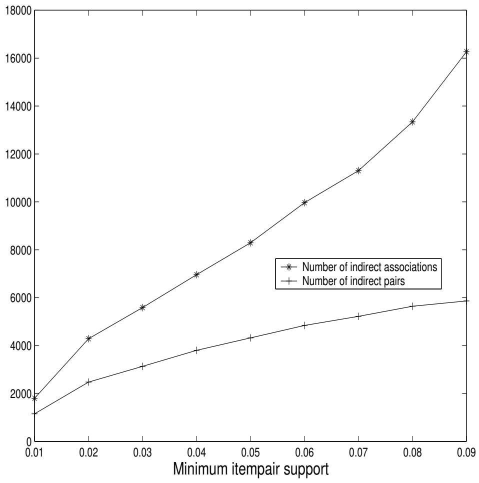

Figure 11: Number of indirect associations and indirect pairs generated for various t_s thresholds (with $t_f = 0.1\%$ and $t_d = 0.10\%$.

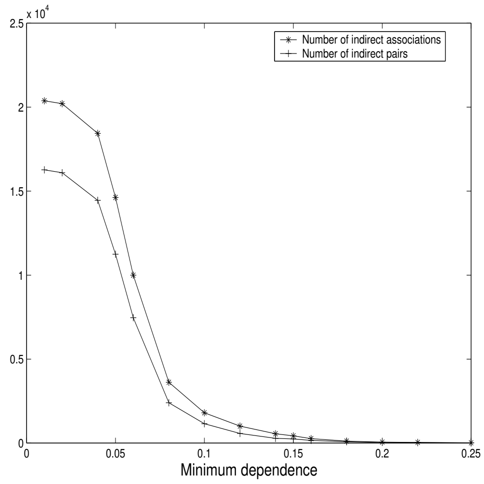

Figure 12: Number of indirect associations and indirect pairs generated for various t_d thresholds (with $t_f = 0.1\%$ and $t_s = 0.01\%$.

[8] R. Cooley, P.N. Tan, and J. Srivastava. Discovery of interesting usage patterns from web data. In M. Spiliopoulou and B. Masand, editors, *Advances in Web Usage Analysis and User Profiling*, volume 1836, pages 163–182. Lecture Notes in Computer Science, 2000.

[9] Robert Cooley, Bamshad Mobasher, and Jaideep Srivastava. Web mining: Information and pattern discovery on the world wide web. In *International Conference on Tools with Artificial Intelligence*, pages 558–567, Newport Beach, 1997. IEEE.

[10] M. Deshpande and G. Karypis. Selective markov models for predicting web page access. In *Proc. of First SIAM Intl Conf on Data Mining*, Chicago, 2001.

[11] M.N. Garofalakis, R. Rastogi, and K. Shim. Spirit: Sequential pattern mining with regular expression constraints. In *Proc. of the 25th VLDB Conference*, pages 223–234, Edinburgh, Scotland, 1999.

[12] B. Liu, W. Hsu, and Y. Ma. Pruning and summarizing the discovered associations. In *Proc. of the Fifth Int'l Conference on Knowledge Discovery and Data Mining*, San Diego, CA, 1999.

[13] H. Mannila, Toivonen H., and A.I. Verkamo. Discovery of frequent episodes in event sequences. *Data Mining and Knowledge Discovery*, 1(3):259–289, 1997.

[14] J. Pei, J. Han, B. Mortazavi-Asl, and H. Zhu. Mining access patterns efficiently from web logs. In *PAKDD 2000*, April 2000.

[15] J.E. Pitkow and P. Pirolli. Mining longest repeating subsequences to predict world wide web surfing. In *USENIX Symposium on Internet Technologies and Systems*, 1999.

[16] A. Savasere, E. Omiecinski, and S. Navathe. Mining for strong negative associations in a large database of customer transactions. In *Proc. of the 14th International Conference on Data Engineering*, Orlando, Florida, February 1998.

[17] C. Shahabi, A.M. Zarkesh, J. Adibi, and V. Shah. Knowledge discovery from users web-page navigation. In *Workshop on Research Issues in Data Engineering*, Birmingham, England, 1997.

[18] A. Silberschatz and A. Tuzhilin. What makes patterns interesting in knowledge discovery systems. *IEEE Trans. on Knowledge and Data Engineering*, 8(6):970–974, 1996.

[19] M. Spiliopoulou, L.C. Faulstich, and K. Winkler. A data miner analyzing the navigational behaviour of web users. In *Proc. of the Workshop on Machine Learning in User Modelling of the ACAI'99 Int. Conf.*, Creta, Greece, July 1999.

[20] R. Srikant and R. Agrawal. Mining sequential patterns: Generalizations and performance improvements. In *Proc. of the Fifth Intl Conf. on Extending Database Technology (EDBT)*, Avignon, France, 1996.

[21] J. Srivastava, R. Cooley, M. Deshpande, and P.N. Tan. Web usage mining: Discovery and applications of usage patterns from web data. *SIGKDD Explorations*, 1(2):12–23, 2000.

[22] P.N. Tan and V. Kumar. Interestingness measures for association patterns : A perspective. In *KDD 2000 Workshop on Postprocessing in Machine Learning and Data Mining*, Boston, MA, August 2000.

[23] P.N. Tan and V. Kumar. Discovery of web robot sessions based on their navigational patterns. *accepted for the special issue of the International Journal of Data Mining and Knowledge Discovery on Web Mining for E-commerce*, 2001.

[24] P.N. Tan, V. Kumar, and Jaideep Srivastava. Indirect association: Mining higher order dependencies in data. In *Proc. of the 4th European Conference on Principles and Practice of Knowledge Discovery in Databases*, pages 632–637, Lyon, France, 2000.

[25] H Toivonen, M. Klemettinen, P. Ronkainen, K. Hatonen, and H. Mannila. Pruning and grouping discovered association rules. In *ECML-95 Workshop on Statistics, Machine Learning and Knowledge Discovery in Databases*, 1995.

[26] A. Wexelblat. An environment for aiding information-browsing tasks. In *Proc. of AAAI Symposium on Acquisition, Learning and Demonstration: Automating Tasks for Users*, Birmingham, UK, 1996.