Mining Uncertain Web Log Sequences with Access History Probabilities [*]

Olalekan Kadri
School of Computer Science
University of Windsor
Windsor, Ontario N9B 3P4
woddlab@uwindsor.ca

C.I. Ezeife[†]
School of Computer Science
University of Windsor
Windsor, Ontario N9B 3P4
cezeife@uwindsor.ca

ABSTRACT

This paper proposes (1) modeling uncertainty in web log sequences using the most recent periodic web log which attaches computed existential probabilities between 0 and 1, to events in the sequences, (2) using the newly proposed uncertain PLWAP web sequential miner for these uncertain access sequences. While PLWAP only considers a session of web logs, U-PLWAP takes more sessions of web logs from which existential probabilities are generated and there is the need to traverse each suffix tree from the root in order to scan for existential probabilities of items already found along the path. Experiments show that U-PLWAP is faster than U-Apriori, and UF-growth.

Categories and Subject Descriptors

H.2.8 [**Database Management**]: [Database Applications, Data Mining]; K.6.5 [**Management of Computing and Information Systems**]: Data Quality and Uncertainty—*Incomplete Data, Prediction*

General Terms

Uncertain Data

Keywords

Uncertain data mining, frequent sequential patterns, web log mining, existential probability generation, dirty data mining

1. INTRODUCTION

In an uncertain sequence database, each event e_{ik} in sequence S_i carries with it the existential probability p_k of the event e_{ik} occurring in S_i. This is written as: e_{i1}:p_1, e_{i2}:p_2, e_{i3}:$p_3 \ldots e_{ik}$:p_k. An example of uncertain sequential database with schema (User ID, Uncertain Access Sequence) to be mined is:

10	(a:1,b:0.5,c:0.75,d:0.5)
20	(a:1,d:0.5,e:0.5,d:0.5)
30	(a:1,b:0.5,d:0.5,b:0.5)

In this uncertain database, although the sequence of events by user id 10 is given as *abcd*, it could be only *ac*, *a*, *ab*, *ad* or other combinations because, the event *a* is the only event with 100% certainty of being in this sequence, while event c also has 75% certainty and events b and d have 50% certainty of being there. In the web log domain as in many other domains, this paper argues that 1) Historical access log sequences D can be transformed into uncertain access sequences like those above, by partitioning the historical access log into a number (n) of time periodic logs $D_1 \ldots D_n$, and the most recent log D_n is used to determine the user's most recent access sequence pattern and the rest of the historical logs ($D_1 \ldots D_n$) are used to determine the certainty (existential probability) of each of the events in the most recent log D_n. 2) The most recent log D_n modeled with each event having an existential probability, is the uncertain database sequence that is mined for frequent patterns using an extended sequential pattern approach for handling uncertain data. From the segmented access history databases, the user access history database D_{user} which shows the history, based on user id is obtained as shown in Table 1. The user access history database D_{user} has possibly, between 1 to n sequences for each user id when the nth sequence is the most recent sequence for the user drawn from the batched access history database.

Table 1: The User Access History Database D_{user}

$UserID$	Time	Sequence
10	4	(a,b,c,d)
	3	(a,d,a,b)
	2	(a,d,c,a)
	1	(a,c,a,c)
20	2	(a,d,e,d)
	1	(a,c,b)
30	4	(a,b,d,b)
	3	(a,c,e,b)
	2	(a,e,d,e)
	1	(a,e)

The existential probability of each event e, for each user

[*]

[†]This research was supported by the Natural Science and Engineering Research Council (NSERC) of Canada under an Operating grant (OGP-0194134) and a University of Windsor grant.

SAC'11 March 21-25, 2011, TaiChung, Taiwan.
Copyright 2011 ACM 978-1-4503-0113-8/11/03 ...$10.00.

u, P_e^u, in the most recent log database is given as: $P_e^u =$ (Number of records (sequences) of u in D_{user} containing e) / (Total number of records of u in D_{user}). Finally, to arrive at the uncertain database sequence used for mining, our proposed approach uses the most recent sequence from the database log D_i accessed by each user to represent the user's sequence in the uncertain sequential database to be mined. Next, the user's access history sequences are used to compute the existential probability of each event in the user's sequence. Thus, our approach accepts each user's most recent visit pattern (drawn from their sequence in the most recent log) as their current sequential pattern (in the uncertain database to be mined) since that reflects the most recent access interests of the user. The existential probability (or likelihood) of each event (item, e.g., a visit to a web site) in these user's most recent sequence is calculated from their historical sequential visits (from their n last sequences obtained from the database logs D_1 to D_n). The likelihood of having the latest log represent a true reflection of how each user browses the site based on the historical behavior is calculated with the equation above for computing the existential probability for each event in the user's sequence P_e^u.

1.1 Contributions and Outline

The contributions of the proposed U-PLWAP approach are as follows. 1. The most current web log access sequence history is used to model the uncertain web log access sequence to be mined by computing from all history logs the existential probabilities of each event in this most current log for each user. 2. Each node representing items in U-PLWAP tree records item's label, occurrence count, position code and the set of existential probability values (more than one) for this item or event. 3. Sequences that share the same prefix/suffix but differ in existential probability of constituent items are combined into same node, making the U-PLWAP tree more compact, and faster to mine. 4. The mining process is done on each suffix tree by dynamically generating cumulative product sequence when new temporary nodes are found.

2. THE PROPOSED U-PLWAP MINING ALGORITHM

Problem Definition: Given an uncertain database sequence UDB obtained after pre-processing as described in contribution 1 above, and a minimum support threshold λ, the task is to find all frequent patterns with support count greater than or equal to the minimum support.

Proposed Problem Solution: The proposed U-PLWAP algorithm goes through the following steps to find frequent sequences as summarized in the formal algorithms provided in Algorithms 1 and 2.

ALGORITHM 1. *(The U-PLWAP Main Algorithm)*

```
Algorithm UPLWAP()
Input:   Uncertain Web Access Sequence Database (UDB),
         minimum support $\lambda(0 < \lambda \leq 1)$.
Output:A list of frequent patterns Fps in UDB.
begin
1. Find the frequent 1-items as described in step 1 of section 3.1.
2. Insert frequent 1-items in the Link header table (section 3.1)
3. Build U-PLWAP tree from the UDB as described in section 3.1.
4. Recursively mine the U-PLWAP tree by calling UPLWAP-Mine
algorithm. end
```

ALGORITHM 2. *(The UPLWAP_Mine Algorithm)*

```
Algorithm UPLWAP_Mine()
Input:   U-PLWAP tree T, Root_set R, Link header table L,
         set of cumulative Products of existential probabilities K,
         Frequent n-sequence F, minimum support $\lambda(0 < \lambda \leq 1)$,
         (R contains root, K and F are empty when the
         algorithm is called the first time).
Output: Frequent n-sequence F'.
Other variables: S stores the information of node whether it is
         the ancestor of the following node in the queue of
         similar nodes, C stores the total count of event $e_i$
         in different suffix tree.
begin
1. If R is empty, return
2. For each event $e_i$ in L, find the suffix tree of $e_i$
in T ($e_i$|suffix tree)
2.1 Save first event in $e_i$-queue to S
2.2 Following the $e_i$-queue,
If event $e_i$ is the descendant of any event in R, and is not
the descendant of S
Insert node of $e_i$ into new root set $R'$.
2.3 Replace S with $e_i$
2.4 If K is empty
     2.4.1 Add all existential probability entries in node $e_i$ into C
     Enter all entries of existential probabilities in node $e_i$ into
     $K'$, each identified by sequence ID
     Else
     2.4.2 Find products of corresponding entries in K and
     entries of set of existential probability values in node $e_i$.
     Add all the products found to C.
     Enter each product of existential probability values in node $e_i$
     into K', each identified by sequence ID.
2.5 If C is greater than support count $\lambda$
     2.5.1 Append $e_i$ to end of F to form $F'$ and output $F'$.
     Call algorithm UPLWAP_Mine passing R', F' and K'.
end
```

3. CONCLUSIONS AND FUTURE WORK

The U-PLWAP, based on PLWAP algorithm [2] outperforms both U-Apriori ([1]) and UF-growth ([3]), while producing accurate patterns. In building the U-PLWAP tree, similar events sharing same path are combined into one node even when they have different existential probabilities. The U-PLWAP approach also eliminates the need to traverse the various tree paths in order to scan for existential probabilities of all items found from the root. Future work can be based on support counts calculated with conditional probability that is dependent on the probability values of constituent items.

4. REFERENCES

[1] C. Chui, B. Kao, and E. Hung. Mining frequent itemsets from uncertainty data. LNAI, 4426, 47-58, 2007.

[2] C. Ezeife and Y. Lu. Mining web log sequential patterns with position coded pre-order linked wap-tree. *International Journal of Data Mining and Knowledge Discovery (DMKD) Kluwer Publishers*, 10(1):5–38, 2005.

[3] C. Leung, M. Mateo, and D. Brajczuk. A tree-based approach for frequent pattern mining from uncertain data. In *Proceedings of 2008 PAKDD*, pages 653–661, 2008.