International Journal of Applied Mathematics and Computing Science. Volume 3, Issue 4, (2014) 52-60
http://www.cayley-nielson.com/ijamcs

Web-path Mining Based on Time Division Matrix

Zhang Yu-Hua
North China University, Xueyuan Road 3, Taiyuan, Shanxi 030051, China
E-mail: *yuhuazh@yeah.net*

Abstract

RCFA (Repeated Continuous Frequent Access) Paths Algorithm has been a very popular set of algorithms in the fields of web log user preference path excavating. One algorithm that stands out from this set is the CA-mining algorithm for its superiority of the excavation result over the others. However even the CA-mining algorithm has neglected the regularity that broadly exists when users are visiting websites. This essay has put forward an improved CA-mining algorithm by using a “time division matrix” model to pre-process the web log data, turning them to matrices of three time granularities and as inputs of the CA-mining algorithm. Added to this new processing method, the improved CA-mining algorithm is shown by the simulation experiments to be of higher precision ratio and utility value than the previous CA-mining algorithm.

Keywords: time division matrix; CA-mining algorithm; user's preferred visiting path; Web-data excavation.

1. Introduction

Currently, in the field of web-data mining, some relevant mining algorithm based on records of users’ browsing paths of web pages has been a heat, among all the new algorithms that has been raised. Some examples of these heated algorithms are FP-Tree (Frequent Pattern Tree) algorithm, mining algorithm based on web-visiting matrix yielded from browsing preferences of users and an algorithm based on RCFA (Repeated Continuous Frequent Access) Paths. These RCFA-based algorithms have been a popular study objects in academies. In particular, the CA-mining algorithm, that is the RCFA path algorithm based on CA matrix[1], has won its credits among academies according to its high efficiencies and satisfying mining results over huge data amounts.

Despite of these, there lies a common flaw of these mentioned algorithms that all of them have neglected the regularity of users’ visiting paths, the regularity of which appears important in that the preferred visiting paths are different according to varied time periods and moments. For example: users are prone to visit some new and information website or information portal website in the morning, whereas they might visit some entertainment websites in the afternoon. Within a week, difference of the preferred paths also lies in whether the day is weekday or weekend day. When doing user browsing preference analysis, if no preprocessing is done to the user’s visiting log of the year, it means that a material rich in information has been abandoned, and the user's preferred visiting path gained in this way will be inaccurate, hence leads to low accuracy of this algorithm.

This essay puts forward a new model based on the time division matrix to make up for the flaw of existing CA-mining algorithm, which means a preprocessing of CA-mining’ s target information[2]. The simulation shows that the new model (with a preprocessing done) has indeed added accuracy to user's preferred visiting path, which is the result of CA-mining[3] algorithm. This shows the effectiveness of the new model raised.

ISSN 2333-2425 /
December, 2014

2. Time Division Matrix

This essay has put forward a concept of "time division matrix", which will be further illustrated as followed. Since most users have varied visiting habits from time to time over a year, we are going to organize the time information in the web logs into three dimensions: month, week and time period. In this way, we have got three time granularities from the web log. The detailed process is given below.

Firstly, through the server, we can get relevant information of users' visiting time, according to which the corresponding time segment of months, weeks and time periods can be achieved; the three time granularities are then converted to a user visiting matrix; three appropriate matrix weighting coefficients are then chosen, which will be used to calculate the weighted average matrix; a "time division matrix" containing the information of month, week and time period is then obtained. Based on the matrix, flexible adjustments can be made to the three weighting coefficients to access some other differently-focused matrices. Through the simulation results, we discover that if a user holds a web-visiting habit that is more regular, then the information we get from his/her corresponding time division matrix will be more close to his/her true web-visiting preferences.

3. Continuous Access (CA) Matrix Introduction

A. Definition

Definition 1. The meaning of differentiating web affairs according to the largest forward references is that an affair is a path from the first page of a forward browsing to the previous page before backspacing in a user conversation.

Definition 2. Suppose $L = \{L_1, L_2, \cdots\cdots L_n\}$ is a path that repeatable and successive visit can be paid to, $D = \{d_1, d_2, \cdots\cdots d_n\}$ is database of accessing sequence. If $\exists i$ makes d_i contain L, then we denote $d_i(L)$, and $|d_i(L)|/|D| = \min\text{supp}$ is the minimum support，then we call L a path that repeatable, successive and frequent visit can be paid to, among which $|d_i(L)|$ and $|D|$ stands for the number of elements respectively, with the minimum support $\partial = \min\text{supp} \times |\mathrm{D}|$.

Definition 3

$L = \{L_1, L_2, \cdots L_i, L_{i+1}, L_n\}$ is a visiting sequence, we call L_{i+1} the direct post-drive URL of L_i in this visiting sequence.

Definition 4

There exist vectors X and Y, if $X_i > 0$, $Y_i > 0$, then currently $X_i = 1$, $Y_i = 1$, hence $X, Y \in \{0,1\}^n$, then the hamming distance between X and Y: $H_d(X, \mathrm{Y}) = \sum_{i=1}^{|\mathrm{x}|} |X_i - Y_i|$.

Definition 5

Continuous Access (CA) Matrix[4]

We set the rows and columns as URL, matrix element A_{ij} is composed of a two-variable set, i.e. $A_{ij} = (V_{ij}, DB_j)$, among which V_{ij} stands for the time that $< V_i, V_j >$ is included in every visiting sequence, DB_j stands for the set of the direct post-drive URL of V_j in every visiting sequence. A matrix that is composed in this way is called a user continuous access matrix (CA).

B. The setup procedure of CA-mining algorithm

The advantage of CA-mining algorithm lies in the fact that it considers both the continuity of the paths and some characteristics of the repeatable paths, based on which an excavation of frequent paths using CA

matrix is conducted [5].

The CA-mining algorithm flow

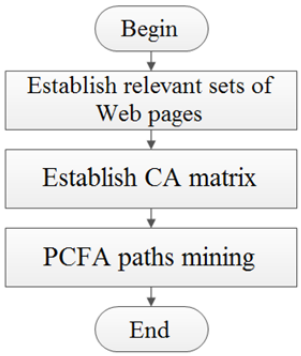

Figure 1. The CA-mining algorithm flow

Pseudo-code description of CA-mining algorithm

1) input：CA matrix, minimum support ∂ , two-variable set tSet .

2) output：a set of frequently-visited paths fre_path_set.

3) Establish stack Stack;

4) Do{ path_set=get_two_set(tSet); // a two-variable

// set is taken out from a

// set of two-variable sets

5) fir_set=get_fir_set(path_set); // take the

//antecedent and succedent

6) beh_set=get_beh_set(path_set);

7) For(int i=0;i<CA;i++){

8) If(CA[i][0]!=fir_set) continue;

9) For(int j=0;j<CA[i].length;j++){

10) If(CA[i][j]==beh_set){

11) If(CA[i][j].Vij<∂) break the two loop;

12) Else{

13) If(path_set is two path&&does not repeat under fre_path_set)

14) Fre_path_set.add(path_set);

15) If(get_equal_DBCount(i,j)>= ∂){

16) Stack.push(fir_set); // save <Vi,Vj> in stack //and remove it from pages

17) Stack.push(beh_set);

18) DBPage=get_DB(i,j);

19) path_set+=DBPage;

20) fir_set=beh_set; beh_set=DBPage;

21) i=0; j=0;}

```
22)          Else {
23)            fer_path_set.add(path_set);
24)          If(Stack!=Empty){
25)        beh_set=Stack.pop();
26)        fir_set=Stack.pop();
27)        Else    break the two loop;
28)        path_set.delete_last_item(); }}}}        //    delete the last //term of path_set
29)        If(tSet==Empty) break;
30)        Else    Stack.clear();}while(True)
```

4. An Improved CA-mining Algorithm Based On Time Division Matrix

As is mentioned in introduction, if we directly excavate the data from a year-long web log without any pre-processing, the accuracy of the excavation result will be badly affected.

Aiming at this shortcoming, this essay has put forward an improved CA-mining algorithm based on time division matrix, which will divide the web log information into three time granularities according to time division matrix, and then convert them into corresponding user visiting matrices. When users are visiting websites, we take out the user visiting matrix of that time from the server, and varied weighted average user visiting matrix will be gained based on different weighting coefficients we choose. The weighted average matrices are then loaded into CA-mining algorithms, which yields different user preferred visiting paths accordingly. This advanced algorithm can yield customized user-preferred visiting paths with satisfying flexibility and accuracy [6].

A. Algorithm flow of advanced CA-mining algorithm based on “time division matrix” model

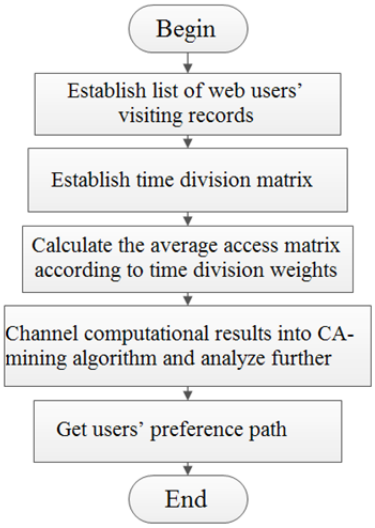

Figure 2. The flow of advanced CA-mining algorithm based on time division matrix

This algorithm firstly makes use of information extracted from users' visits, which is then disposed using time division. The visiting time is divided into measures of three time granularities, “month, week, period”, and the information is then written to these three corresponding files, which is then converted into users' visiting matrices. Thus, we can get the time division matrices when users are visiting websites, and after choosing some weighting coefficients, users' averaged visiting matrices M can be obtained. Put M as

an input into the CA-mining, do the excavation work and so much so the user-preferred visiting paths can be analyzed.

B. Explanation of key algorithm procedures

Establish visiting matrix of web users

Make records of Time in the list according to web users' visiting[7]. Divide the time according to three time granularities of "month, week, time period". Establish array of visiting matrices, among which $A[i]$, $(\forall i, i = 1,2,\cdots 12)$ shall be used to represent 12 months; $B[i]$, $(\forall i, i = 1,2\cdots 7)$ shall be used to represent 7 days in a week; $C[i]$, $(\forall i, i = 1,2,3)$ shall be used to represent three typical time periods in a day, that is 8:00~12:00, 12:00~18:00, 18:00~ 8:00 in the next day. Every visiting matrix holds the same structure: rows stands for website of *OldURL*; columns stands for websites of *NewURL*; the value of elements are the number of times that users skip from *OldURL* to *NewURL*. A NULL value should be added to both rows and columns of the visiting matrices, which in the row vectors, stands for that users are visiting the website via typing in URL, label visiting or through other websites' linkage instead of webpage linkage; in the column vectors, this stands for that users have closed this webpage or stop visiting this webpage.

When we are reading the information in a web log, we firstly convert the Time in records into month m, weekday w and time period d, then convert the *OldURL* and *NewURL* in the records into row and columns in the matrix; at last, we let the values in matrix $A[m]$, $B[w]$, $C[d]$ be added to 1, so on and so forth, we will get the final execution results through iteration.[8]

Calculate the average visiting matrix M

Firstly, by using the visiting information of time, we calculate the month of visiting time m, week w and time period d, then take out the three elements $A[m]$, $B[w]$, $C[d]$ from the matrix; by using the formula $M = \alpha A[m] + \beta B[w] + \gamma C[d]$, we can calculate the average visiting matrix M, among which the weighting coefficients α, β, γ can be determined by experimental parameter. By adjusting the proportion of α, β, γ, we can get user visiting matrices with good illustrating effects.

The description of the improved CA-mining algorithm[9] based on "time division matrix" is as below:

1) input: web log Rec; weighting coefficients of month, week and time period in a day: mm, ww, dd, threshold p
2) output: the set of users' preferred visiting paths
3) Establish the matrix array of month, week and time period in a day A, B, C; the initial matrix is 0.
4) For each rec Rec
5) Do{
6) If rec.State=200 and rec.Method='Get' Then
7) month=get_month(rec.Time);
8) week=get_week(rec.Time);
9) daytime=get_daytime(rec.Time);

```
10)              row=get_row(rec.OldURL);
11)              column=get_col(rec.NewURL);
12)            A[month](row,column)=A[month](row,column)+1;
13)              B[week](row,column)=B[week](row,column)+1;
14)              C[time_a_day](row,column)=C[time_a_day](row,column)+1;
15)          End If
16)          End For
17)          cur_month=data_time.month;
18)          cur_week=data_time.week;
19)          cur_time=data_time.time_a_day;
20)          For i=0 to N
21)          For j=0 to N
22)    M(i,j)=mm*A[cur_month](i,j)+ww*B[cur_week](i,j)+dd*C[cur_time](i,j);
23)          End For
24)          End For}
25)          Call CA-mining(M , p);
```

The algorithm executes as follows. Firstly, it reads the users' visiting records from a current database table Rec and get Time in each record. Then Time is converted into corresponding month month、week week and time period of the day time_a_day; then OldURL and NewURL is converted to row and column of the matrix. Then, we go through the whole Rec and add 1 to the value of A[month], B[week], C[time_a_day]. We will then obtain the corresponding month cur_month、week cur_week and time period cur_time. The average matrix M will then be obtained by entering the weighting coefficients of the matrix. At last, import M and p into CA-mining algorithm to get the set of users' preferred visiting path [10].

5. Simulation Experiment

In order to demonstrate the accuracy that the improved algorithm owns on excavating the preferred path of the web users, we have used the same data source and simulated the process by two ways—one adopts the time division matrix model while the other one does not. The accuracy is calculated in two ways.

5.1. Simulation environment

Operating system：Windows 7

Machine configuration：2G RAM， CPU 2.20GHz

Experimental tools：MySQL 2008 database， Eclipse 3.7.2

Data source：IIS web log data (test-purpose) from a noted microblog company

5.2. Definition of precision ratio

In order to test that whether the time division matrix model has improved the CA-mining algorithm and to represent the satisfying degree by users to the preferred web-visiting path given to them, we have raised the concept of precision ratio. Precision ratio is the reflect of whether customers are satisfied with the preferred web-visiting path given to them by the website.

The detailed definition of precision ratio is as follows.

Suppose the set of preferred web-visiting path yielded based on the time division matrix model is

$T, T = \{T_i \mid i = 2,3 \cdots N\}$, among which T_i stands for the subset of preferred path whose length is i, N stands for the largest length of users' preferred path. Define $R_i = |T_i|$, which stands for the number of elements that the set of preferred path has. We assume that users choose a provided preferred visiting path at random. Thus, the probability of user's choosing a preferred visiting path from a subset T_i is $P_i = 1/R_i$, and the precision ratio v is $v = (\sum_{i=2}^{N} 1/\mathrm{R}_i) \times [1/(\mathrm{N}-1)]$

5.3. Simulation process

Suppose that users visit websites with certain regularity. To strengthen the simulation effect, we ask users (subjects) to visit websites with strong regularity, that is, they have fixed preferred websites to visits in the morning, afternoon and at night. By setting different URL numbers, we obtained the data based on time division matrix model. Simulation is done under the condition when the number of URL is 30 40 50 60 70 80.

When $\alpha : \beta : \gamma = 3:3:4$, we get the following result.

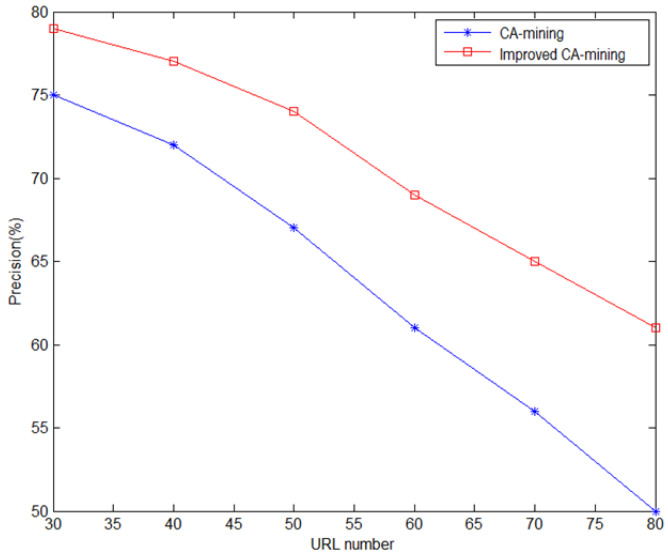

Figure 3. $\alpha : \beta : \gamma = 3:3:4$ result

From the graph we can see that the improved CA-mining algorithm based on time division matrix has better accuracy than before, when no time division model was adopted. What's more, as the number of URL increases, the advantage of this improved algorithm becomes more obvious.

After increasing the proportion of γ, when $\alpha : \beta : \gamma = 2:2:6$, the result is

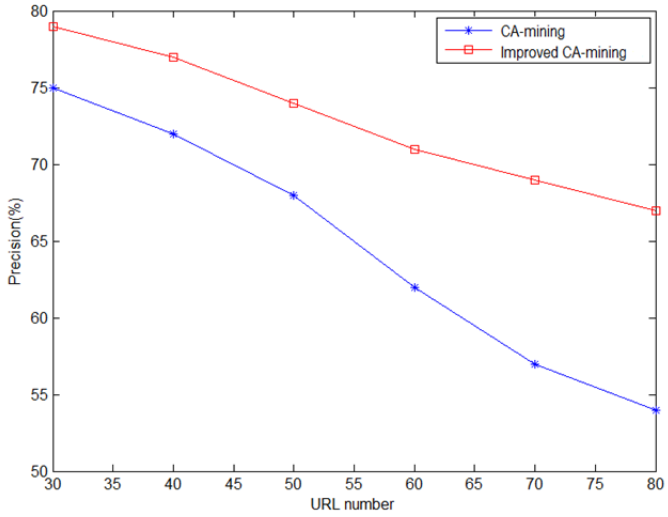

Figure 4. $\alpha:\beta:\gamma=2:2:6$ result

5.4. Analysis of simulation result

The simulation result tells that the improved CA-mining algorithm based on time division matrix has indeed raised the precision ratio when excavating the user-preferred visiting path. Besides, as the change of some parameters, different results yield. As the proportion of some certain time granularity increases, the precision of the algorithm also increases.

6. Conclusion

This essay has raised an improved CA-mining algorithm using time division matrix based on current CA-mining algorithm, which is an RCFA path excavating algorithm of CA matrix. Aiming at excavating relevant information of users' preferred visiting path, the improved algorithm has shown a better precision ratio than the current one in the simulation experiment, which has proved the value and significance of the adoption of a time division matrix pre-processing method used in the improved CA-mining algorithm. The following prospects of research will be searching for efficient ways to popularize this improved CA-mining algorithm in order to excavate user preference data and ways to make better and more precise forecasting and modeling of users' website-visiting behaviors.

References

[1] Srivastava, T., Prasanna Desikan, and Vipin Kumar. "Web mining–concepts, applications and research directions." Foundations and Advances in Data Mining. Springer Berlin Heidelberg, 2005. 275-307.

[2] Joachims, Thorsten, Dayne Freitag, and Tom Mitchell. "Webwatcher: A tour guide for the world wide web." IJCAI (1). 1997.

[3] Ngu, Daniel Siaw Weng, and Xindong Wu. "Sitehelper: A localized agent that helps incremental exploration of the World Wide Web." Computer Networks and ISDN Systems 29.8 (1997): 1249-1255.

[4] Anceaume, Emmanuelle, et al. "P2P architecture for self-atomic memory." Parallel Architectures, Algorithms and Networks, 2005. ISPAN 2005. Proceedings. 8th International Symposium on. IEEE, 2005.

[5] Park, Jong Soo, Ming-Syan Chen, and Philip S. Yu. "Efficient parallel data mining for association rules." Proceedings of the fourth international conference on Information and knowledge management. ACM, 1995.

[6] Sarabjot Singh Anana,Barnshad Mobasher. Introduction to Intelligent Techniques for Web Personalization. ACMTransactions on Intemet Technology, 2007, 7(4):18-22.

[7] Anand, Sarabjot Singh, and Bamshad Mobasher. "Introduction to intelligent techniques for web personalization." ACM Transactions on Internet Technology (TOIT) 7.4 (2007): 18.

[8] Agrawal, Rakesh, and John C. Shafer. "Parallel mining of association rules." IEEE Transactions on knowledge and Data Engineering 8.6 (1996): 962-969.

[9] Mobasher, Bamshad, et al. "Effective personalization based on association rule discovery from web usage data." Proceedings of the 3rd international workshop on Web information and data management. ACM, 2001.

[10] Géry, Mathias, and Hatem Haddad. "Evaluation of web usage mining approaches for user's next request prediction." Proceedings of the 5th ACM international workshop on Web information and data management. ACM, 2003.