Thuật toán khai phá mẫu dãy thường xuyên trọng số chuẩn hóa với khoảng cách thời gian

Mining Normalized Weighted Frequent Sequential Patterns with Time Intervals Algorithm

Trần Huy Dương, Vũ Đức Thi

Abstract: *In this paper, we propose a method for mining normalized weighted frequent sequential patterns with time intervals, we are not only interested in the number of occurrences of the sequence (the support), but also concerned about their levels of importance (weighted). We use the binding between the support and weight of the set range to candidates in mining normalized weighted frequent sequential patterns with time intervals while maintaining the downward closure property nature which allows a balance between support and the weight of a sequence.*

Keywords*: Data mining, frequent sequential patterns, time intervals, weighted, sequential patterns.*

I. GIỚI THIỆU

Khai phá mẫu dãy (Mining Sequential Patterns) là một trong những lĩnh vực rất quan trọng trong nghiên cứu khai phá dữ liệu và được áp dụng trong nhiều lĩnh vực khác nhau. Trong thực tế các dữ liệu dãy tồn tại rất phổ biến như dãy dữ liệu mua sắm của khách hàng, dữ liệu điều trị y tế, nhật ký truy cập web, v.v... Mục đích chính của khai phá mẫu dãy là phát hiện tất cả các dãy con xuất hiện lặp lại trong một CSDL theo yếu tố thời gian.

Hiện nay trên thế giới có nhiều nhóm tác giả nghiên cứu đề xuất các thuật toán với các phương pháp tiếp cận khai phá mẫu dãy khác nhau như AprioriAll [1], GSP [2], PrefixSpan [3], SPADE [4], SPAM [5], CloFS-DBV [13] v.v... nhằm giải quyết sự đa dạng của các loại bài toán cũng như đưa ra các hướng cải tiến nhằm giảm thiểu chi phí thời gian và tài nguyên hệ thống.

Các thuật toán kể trên khai phá mẫu dãy chỉ quan tâm đến số lần xuất hiện (hay độ hỗ trợ) của các mẫu dãy; thuật toán do Hirate và Yamana [10] đề xuất cho phép khai phá các mẫu dãy có quan tâm đến giá trị của khoảng cách thời gian giữa các dãy. Tuy nhiên, các thuật toán này cơ bản chưa quan tâm đến sự ràng buộc giữa khoảng cách thời gian giữa các dãy và mức độ quan trọng khác nhau của các mục dữ liệu. Vì vậy, bài báo nhằm đề xuất một phương pháp khai phá mẫu dãy trọng số chuẩn hóa với khoảng cách thời gian, chúng tôi không chỉ quan tâm đến số lần xuất hiện của các dãy (độ hỗ trợ) mà còn quan tâm đến khoảng cách thời gian giữa các dãy và mức độ quan trọng khác nhau (trọng số) của chúng. Chúng tôi sử dụng tính chất ràng buộc giữa độ hỗ trợ, khoảng cách thời gian và trọng số của dãy để sinh các tập ứng viên trong khai phá mẫu dãy thường xuyên trọng số chuẩn hóa với khoảng cách thời gian trong khi vẫn sử dụng tính chất phản đơn điệu cho phép cân bằng độ hỗ trợ, khoảng cách thời gian và trọng số của một dãy.

Phần còn lại của bài báo như sau: Phần II trình bày các nghiên cứu liên quan. Phần III trình bày bài toán và đề xuất thuật toán khai phá mẫu dãy thường xuyên trọng số chuẩn hóa với khoảng cách thời gian WIPrefixSpan dựa trên giải thuật khai phá mẫu dãy thường xuyên PrefixSpan [3] và thuật toán do Hirate và Yamana [10] đề xuất. Phần IV trình bày kết quả thực nghiệm và so sánh giải thuật đề nghị (WIPrefixSpan), WPrefixSpan [11] và giải thuật IPrefixSpan[10] trên bộ dữ liệu BMS-WebView. Kết

luận và hướng phát triển được thể hiện trong phần cuối.

II. CÁC NGHIÊN CỨU LIÊN QUAN

Năm 1995, Agrawal và Srikant phát triển bài toán khai phá mẫu dãy phổ biến [1] và đề nghị thuật toán AprioriAll một thuật toán dựa trên thuật toán Apriori để khai thác mẫu dãy phổ biến. Cũng giống như Apriori, AproiriAll quét CSDL nhiều lần và dựa vào phương pháp sinh ứng viên nên tốn thời gian khai phá.

Năm 2001, Pei và các đồng sự đề nghị thuật toán PrefixSpan [3], một thuật toán dựa trên phương pháp phát triển mẫu dãy. Thuật toán không phải quét CSDL nhiều lần nên thời gian khai phá giảm đi đáng kể so với AprioriAll [1]. Các thuật toán sau đó được phát triển nhằm tối ưu hóa quá trình khai phá mẫu dãy có thể kể đến như SPADE [4], SPAM [5].

Ngoài ra, các kỹ thuật dựa trên chuỗi bit động để khai phá mẫu dãy đóng cũng được đề nghị trong [13].

Đối với khai phá mẫu trên CSDL có trọng số thì các thuật toán khai phá mẫu dãy nêu trên không quan tâm tới mức độ quan trọng của từng mẫu (trọng số của mẫu). Trên thế giới có nhiều tác giả đã nghiên cứu về trọng số, có thể kể đến là các công trình khai phá tập mục có trọng số [6-9], [14-16], mẫu dãy có trọng số [12,15]. Các thuật toán [12,15] tuy khai phá mẫu dãy trên CSDL có trọng số nhưng chưa quan tâm đến các ràng buộc giữa trọng số, độ hỗ trợ và khoảng cách thời gian của dãy.

III. KHAI PHÁ MẪU DÃY THƯỜNG XUYÊN TRỌNG SỐ CHUẨN HÓA VỚI KHOẢNG CÁCH THỜI GIAN

III.1. Các thuật ngữ mô tả bài toán

Cho I = {i_1, i_2, …, i_n} là tập hợp các mục dữ liệu. Mỗi mục $i_j \in$ I được gán một trọng số w_j *với giá trị* $j=1,...,n$.

Một dãy S_m là một danh sách được sắp xếp theo thứ tự của các mục dữ liệu dạng {($t_{1,1}$,s_1), ($t_{1,2}$,s_2), ($t_{1,3}$,s_3),..., ($t_{1,m}$,s_m)} với $s_j \subseteq$ I trong đó ($1 \le j \le m$) là một tập mục được gọi là thành phần của dãy và s_j có dạng ($i_1 i_2 \dots i_k$) và i_t là một mục dữ liệu thuộc I, và $t_{\alpha,\beta}$ là khoảng cách thời gian giữa dãy s_α và s_β. Một dãy S_m bị loại nếu chỉ có duy nhất một mục dữ liệu $i_t \subseteq$ I. Một mục dữ liệu chỉ xuất hiện 1 lần trong thành phần của s_j, nhưng có thể xuất hiện nhiều lần trong các thành phần của một dãy S_m.

Kích thước $|S_m|$ của một dãy là số lượng của các thành phần trong dãy S_m. Độ dài $l(S_m)$ của dãy là tổng số mục dữ liệu trong dãy S_m. Một cơ sở dữ liệu dãy S = {S_1, S_2, …, S_n} là một tập các bộ dữ liệu (sid,S_k) với sid là định danh của một dãy và S_k là một dãy dữ liệu có dạng {($t_{1,1}$,s_1), ($t_{1,2}$,s_2), ($t_{1,3}$,s_3),..., ($t_{1,m}$,s_m)}.

Định nghĩa 1. (Dãy dữ liệu có khoảng cách thời gian): Một dãy dữ liệu có khoảng cách thời gian có dạng:

$$S_m = <(t_{1,1},s_1), (t_{1,2},s_2), (t_{1,3},s_3),..., (t_{1,m},s_m)> \quad (1)$$

Với $t_{\alpha,\beta}$, là khoảng cách thời gian giữa dãy s_α và s_β có dạng:

$$t_{\alpha,\beta} = s_\beta.time - s_\alpha.time \quad (2)$$

Định nghĩa 2. (Độ hỗ trợ của một dãy) : Độ hỗ trợ của một dãy S_a trong một cơ sở dữ liệu dãy S là số lượng xuất hiện các bản ghi trong S có chứa dãy S_a .

Định nghĩa 3. (Trọng số chuẩn hóa của dãy): ChoI={i_1, i_2, …, i_n} là tập hợp các mục dữ liệu. Mỗi mục $i_j \in$I được gán một trọng số w_j, $j = 1,...,n$. Khi đó trọng số chuẩn hóa của một dãy α = <($t_{1,1}$,s_1), ($t_{1,2}$,s_2), ($t_{1,3}$,s_3),..., ($t_{1,m}$,s_m) > có độ dài k và s_j có dạng ($i_1 i_2 \dots i_k$) được tính bằng công thức:

$$NW(\alpha) = \frac{1}{k}\sum_{i_j \in \alpha} w_j \quad (3)$$

Định nghĩa 4. (Độ hỗ trợ với trọng số chuẩn hóa của dãy):

Ta gọi đại lượng *NWsupport(α)* của dãy α là độ hỗ trợ với trọng số chuẩn hóa của dãy α:

$$NWsupport(\alpha) = NW(\alpha) * support(\alpha) = \left(\frac{1}{k}\sum_{i_j \in \alpha} w_j\right) * support(\alpha) \quad (4)$$

Định nghĩa 5. (Ràng buộc khoảng cách thời gian): Cho một dãy < ($t_{1,1}$,s_1), ($t_{1,2}$,s_2), ($t_{1,3}$,s_3),..., ($t_{1,m}$,s_m)>, các ràng buộc khoảng cách thời gian giữa các dãy được định nghĩa theo [10]:

- C_1 = *min_time_interval* là giá trị nhỏ nhất giữa hai dãy liền kề, tức là $t_{i,i+1} \geq$ *min_time_interval.*
- C_2 = *max_time_interval* là giá trị lớn nhất giữa hai dãy liền kề, tức là $t_{i,i+1} \leq$ *max_time_interval.*
- C_3 = *min_whole_interval* là giá trị nhỏ nhất giữa dãy đầu và dãy cuối, tức là $t_{i,m} \geq$ *min_whole_interval.*
- C_4 = *max_whole_interval* là giá trị lớn nhất giữa dãy đầu và dãy cuối, tức là $t_{i,m} \leq$ *max_whole_interval.*

Định nghĩa 6. (Mẫu dãy thường xuyên trọng số chuẩn hóa với khoảng cách thời gian): Cho một CSDL dãy S có khoảng cách thời gian giữa các dãy, mỗi mục $i_j \subseteq$ I được gán một trọng số w_j, một ngưỡng hỗ trợ tối thiểu wminsup, các ràng buộc khoảng cách thời gian C_1, C_2, C_3, C_4**.** Một dãy α được gọi là mẫu dãy thường xuyên trọng số chuẩn hóa với khoảng cách thời gian nếu thỏa mãn tính chất:

NWSupport(α) ≥ wminsup và $t_{\alpha,\beta}$ thỏa mãn tính chất các ràng buộc C_1, C_2, C_3, C_4 (5)

Khi đó bài toán khai phá mẫu dãy thường xuyên trọng số chuẩn hóa với khoảng cách thời gian được phát biểu như sau:

- Cho một CSDL dãy S có khoảng cách thời gian giữa các dãy, mỗi mục $i_j \in$ I được gán một trọng số w_j, một ngưỡng hỗ trợ tối thiểu wminsup, các ràng buộc khoảng cách thời gian C_1, C_2, C_3, C_4. Tìm tất cả các mẫu dãy thường xuyên trọng số chuẩn hóa với khoảng cách thời gian trong S, tức là tìm tập L:

L = {$S_a \subseteq S$ | NWsupport(S_a) ≥ wminsup và $t_{\alpha,\beta}$ thỏa mãn tính chất các ràng buộc C_1, C_2, C_3, C_4} (6)

- Mẫu dãy thường xuyên trọng số chuẩn hóa với khoảng cách thời gian không thỏa mãn tính chất phản đơn điệu, nghĩa là tập con của một mẫu dãy thường xuyên trọng số chuẩn hóa với khoảng cách thời gian không nhất thiết phải là mẫu dãy thường xuyên trọng số chuẩn hóa với khoảng cách thời gian.

III.2. Cơ sở toán học

Chúng tôi đề xuất một thuật toán khai phá mẫu dãy thường xuyên trọng số chuẩn hóa với khoảng cách thời gian (**WIPrefixSpan**), trong đó các Định nghĩa 7, 8, 9 dựa trên giải thuật khai phá mẫu dãy thường xuyên PrefixSpan [3] và của Hirate vàYamana[10] với cách tiếp cận chính là tìm cách đưa các ràng buộc trọng số, ràng buộc khoảng cách thời gian và độ hỗ trợ trong thuật toán khai phá mẫu dãy đảm bảo tính chất phản đơn điệu.

Để tránh phải thực hiện kiểm tra tất cả khả năng kết hợp của một dãy các ứng cử viên tiềm năng, ta có thể thay thế các thứ tự của các mục dữ liệu trong từng thành phần trong dãy. Khi đó các mục trong một thành phần của một dãy có thể được liệt kê theo 1 trật tự mà không mất tính tổng quát và có thể giả định rằng thứ tự này luôn luôn được liệt kê theo thứ tự bảng chữ cái.

Ví dụ như dãy < (0,a) (1,acb) (2,ac) > thể hiện thông tin đi siêu thị của khách hàng, tại thời điểm 0 thì khách hàng mua mặt hàng a, thời điểm 1 thì khách hàng mua các mặt hàng a,c,b và thời điểm 2 thì khách hàng mua các mặt hàng a,c. Khi đó, việc thể hiện dãy ban đầu thành dãy <(0,a) (1,abc) (2,ac)> là tương tự vì thể hiện đúng từng mặt hàng khách hàng đã mua trong từng thời điểm cụ thể. Với quy ước như vậy thì sự biểu hiện của một dãy là duy nhất.

Nếu ta theo thứ tự của các tiền tố của một dãy và CSDL điều kiện của tiền tố đó chỉ có các hậu tố của một dãy thì ta có thể kiểm tra các dãy con theo thứ tự sắp xếp trên và CSDL điều kiện theo tiền tố đó.

Định nghĩa 7. (Tiền tố và Hậu tố trong dãy có khoảng cách thời gian): Các mục dữ liệu trong một thành phần của dãy đã sắp xếp thứ tự chữ cái [3]. Cho một dãy a = <($t_{1,1}$,s_1), ($t_{1,2}$,s_2), ($t_{1,3}$,s_3),..., ($t_{1,m}$,s_m)>, một dãy s_b là một tập các $i_j \subseteq$ I. Khi đó tồn tại một giá trị j ($1 \le j \le m$) sao cho $s_b \subseteq s_j$ và $t_{1,b} = t_{1,j}$.

Ta định nghĩa tiền tố trong dãy có khoảng cách thời gian của a với các giá trị s_b, $t_{1,b}$ như sau:

$$\text{Prefix}(a,s_b,t_{1,b}) = <(t_{1,1},s_1), (t_{1,2},s_2), (t_{1,3},s_3),..., (t_{1,j},s_b)> \quad (7)$$

Khi đó hậu tố trong dãy có khoảng cách thời gian của a với giá trị s_b, $t_{1,b}$ được định nghĩa:

$$\text{Postfix}(a,s_b,t_{1,b}) = <(t_{j,j},s'_j), (t_{j,j+1},s_{j+1}), ..., (t_{j,m},s_m)> \quad (8)$$

Với s'_j là một tập con của s_j sau khi đã trừ đi tập s_b. Khi $s'_j = \varnothing$, thì hậu tố của a với giá trị s_b, $t_{1,b}$ là:

$$\text{Postfix}(a,s_b,t_{1,b}) = <(t_{j,j+1},s_{j+1}), (t_{j,j+2},s_{j+2}), ..., (t_{j,m},s_m)>$$

Mặt khác, khi không tồn tại một giá trị j thì hậu tố của a với các giá trị s_b, $t_{1,b}$ trở thành:

$$\text{Prefix}(a,s_b,t_{1,b}) = \varnothing$$

$$\text{Postfix}(a,s_b,t_{1,b}) = \varnothing$$

Ví dụ :

Cho một dãy a = <(0,p),(1,pqr),(3,pr)>, khi đó

Với tiền tố s_b= (0,p) ta xây dựng các hậu tố của dãy a như sau:

Với j= 1, thì $s'_1 = \varnothing$, vì vậy hậu tố là

$$\text{Postfix}(a,s_b,t_{1,b}) = <(1,pqr),(3,pr)>$$

Với j= 2, thì s'_2= (0,qr), vì vậy hậu tố là

$$\text{Postfix}(a,s_b,t_{1,b}) = <(0,qr),(2,pr)>$$

Với j= 3, thì s'_3= (0,r), vì vậy hậu tố là

$$\text{Postfix}(a,s_b,t_{1,b}) = <(0,r)>$$

Định nghĩa 8. (CSDL điều kiện theo tiền tố trong dãy có khoảng cách thời gian): Cho một cơ sở dữ liệu dãy S có khoảng cách thời gian và một dãy b có dạng<($t_{1,1}$,s_1), ($t_{1,2}$,s_2), ($t_{1,3}$,s_3),..., ($t_{1,m}$,s_m)>. Khi đó một CSDL điều kiện theo tiền tố b được định nghĩa là $S|_b$, gồm các hậu tố (Postfix) của các dãy trong S với tiền tố b được xây dựng theo Định nghĩa 7.

Dựa trên Định nghĩa 3, có thể thấy NW(α) luôn nhỏ hơn hay bằng MaxW với MaxW là giá trị lớn nhất của trọng số trong các mục trong S. Vì vậy, thay vì khai phá mẫu dãy thường xuyên thỏa Định nghĩa 6, ta đưa bài toán về dạng khai thác các mẫu dãy thỏa Định nghĩa 9 dưới đây, sau đó tính giá trị ***NWSupport*** của các mẫu dãy thu được để tìm các mẫu dãy thường xuyên.

Định nghĩa 9. (Mẫu dãy ứng viên): Cho một ngưỡng hỗ trợ tối thiểu wminsup. Một dãy α được gọi là dãy ứng viên mẫu dãy thường xuyên trọng số chuẩn hóa với khoảng cách thời gian nếu thỏa mãn tính chất:

Support(α) * MaxW $\ge$ wminsup và thỏa mãn các tính chất C_1, C_2, C_3, C_4 (9)

Với MaxW là giá trị lớn nhất của trọng số trong các mục trong S, mẫu dãy ứng viên được xây dựng nhằm mục đích tỉa bớt không gian tìm kiếm mà vẫn đảm bảo tính phản đơn điệu trong khai phá mẫu dãy thường xuyên trọng số chuẩn hóa với khoảng cách thời gian.

III.3. Ví dụ khai phá mẫu dãy thường xuyên trọng số chuẩn hóa với khoảng cách thời gian

Cho một CSDL dãy S với khoảng cách thời gian như Bảng 1, giá trị các trọng số của các mục dữ liệu như Bảng 2, giá trị wminsup =1,5 và các giá trị C_1=1; C_2=2; C_3=2; C_4=3. Khi đó việc khai phá các mẫu dãy thường xuyên trọng số chuẩn hóa với khoảng cách

thời gian trong CSDL dãy S theo phương pháp WIPrefixSpan được thực hiện theo các bước như sau:

Bảng 1. *Cơ sở dữ liệu dãy S*

iSID	Dãy dữ liệu
10	<(0,a),(1,abc),(3,ac)>
20	<(0,ad),(3,c)>
30	<(0,aef),(2,ab)>

Bảng 2. *Giá trị trọng số của các mục dữ liệu*

Tên mục	Trọng số
a	0.9
b	0.75
c	0.8
d	0.85
e	0.75
f	0.7

Bước 1: Tìm các ứng viên của mẫu dãy thường xuyên với trọng số chuẩn hóa có độ dài 1

Duyệt CSDL dãy S lần đầu tiên để tìm tất cả các ứng viên của mẫu dãy thường xuyên trọng số chuẩn hóa với khoảng cách thời gian có độ dài 1, thực hiện đếm độ hỗ trợ của mỗi mục.

Một mục có độ dài 1 có thể không phải là một mẫu dãy thường xuyên có trọng số chuẩn hóa nhưng có thể kết hợp với các mục khác có độ hỗ trợ lớn hơn hoặc trọng số lớn hơn để trở thành mẫu dãy thường xuyên có trọng số trong các mẫu có độ dài lớn hơn.

Khi đó ta có các giá trị độ hỗ trợ của mỗi mục như sau:

support(<0,a>) = 3, support(<0,b>) = 2,
support(<0,c>) = 2, support(<0,d>) = 1,
support(<0,e>) = 1, support(<0,f>) = 1

Giá trị trọng số các mục theo Bảng 2 là (a=0,9; b=0,75; c=0,8; d=0,85; e=0,75; f=0,7)

Giá trị MaxW = 0.9; Giá trị wminsup = 1,5

Kiểm tra theo Định nghĩa 9, loại mục <0,d>, <0,e> và <0,f> do giá trị:

support(<0,d>)*MaxW = 1*0,9=0,9 < wminsup;

support(<0,e>)*MaxW = 1*0,9=0,9 < wminsup;

support(<0,f>)*MaxW = 1*0,9=0,9 < wminsup;

Khi đó ta có các ứng viên mẫu dãy thường xuyên trọng số chuẩn hóa với khoảng cách thời gian có độ dài 1 là:

$$Q_1 = <0,a>, <0,b>,<0,c>$$

Kiểm tra theo Định nghĩa 6 với các ứng viên trong Q_1, khi đó ta có các mẫu dãy thường xuyên trọng số chuẩn hóa với khoảng cách thời gian có độ dài 1 được nạp vào L là:

NWsupport(<0,a>) = 3*0,9 =2,7 > wminsup;

NWsupport(<0,b>) = 2*0,75 =1,5 = wminsup;

NWsupport(<0,c>) = 2*0,8 =1,6 > wminsup.

Kết quả L tại Bước 1 là :

L = {<0,a>, <0,b>,<0,c>}

Bước 2: Chia không gian tìm kiếm

Toàn bộ các ứng viên và mẫu dãy thường xuyên trọng số chuẩn hóa với khoảng cách thời gian được khai phá trong các phân vùng gồm 03 vùng tương ứng với 03 tiền tố gồm:

- Mẫu dãy với tiền tố <0,a>
- Mẫu dãy với tiền tố <0,b>
- Mẫu dãy với tiền tố <0,c>

Bước 3: Khai phá các tập con ứng viên và mẫu dãy thường xuyên trọng số chuẩn hóa với khoảng cách thời gian

Các tập con ứng viên và mẫu dãy thường xuyên trọng số chuẩn hóa với khoảng cách thời gian được khai phá bằng cách xây dựng các CSDL điều kiện

tương ứng với các tiền tố và khai phá chúng bằng phương pháp đệ quy. Các bước thực hiện như sau:

A. *Tìm các ứng viên và các mẫu dãy thường xuyên trọng số chuẩn hóa với khoảng cách thời gian với tiền tố <0,a> và các ràng buộc thời gian C_1, C_2, C_3, C_4*

Khi đó CSDL điều kiện với tiền tố <0,a> sẽ bao gồm các hậu tố được xây dựng theo Định nghĩa 7:

Bảng 3. *Cơ sở dữ liệu điều kiện với tiền tố <0,a>*

iSID	Dãy dữ liệu
10	<(1,abc),(3,ac)> <(0,bc),(2,ac)> < (0,c)>
20	<(0,d),(3,c)>
30	<(0,ef),(2,ab)> <(0,b)>

Bằng cách quét CSDL điều kiện với tiền tố <0,a>, độ hỗ trợ của các mục dữ liệu của nó là:

support(<1,a>)= 1; support(<1,b>)= 1;
support(<1,c>)= 1; support(<3,a>)= 1;
support(<3,c>)= 2; support(<0,b>)= 2;
support(<0,c>)= 1; support(<2,a>)= 2;
support(<2,c>)= 1; support(<0,d>)= 1;
support(<0,e>)= 1; support(<0,f>)= 1;
support(<2,b>)= 1;

Kiểm tra theo Định nghĩa 9, tìm các mẫu ứng viên có độ dài 2 với tiền tố <0,a>:

support(<1,a>)*MaxW = 1*0,9=0,9
support(<1,b>)*MaxW = 1*0,9=0,9
support(<1,c>)*MaxW = 1*0,9=0,9
support(<3,a>)*MaxW = 1*0,9=0,9
support(<3,c>)*MaxW = 2*0,9=1,8
support(<0,b>)*MaxW = 2*0,9=1,8
support(<0,c>)*MaxW = 1*0,9=0,9
support(<2,a>)*MaxW = 2*0,9=1,8
support(<2,c>)*MaxW = 1*0,9=0,9
support(<0,d>)*MaxW = 1*0,9=0,9
support(<0,e>)*MaxW = 1*0,9=0,9
support(<0,f>)*MaxW = 1*0,9=0,9
support(<2,b>)*MaxW = 1*0,9=0,9

Các ứng viên có độ dài 2 với tiền tố <0,a> thỏa mãn độ hỗ trợ với trọng số lớn nhất là :

<0,ab>, <(0,a),(2,a)>,<(0,a),(3,c)>

Kiểm tra các ứng viên trên với tính chất ràng buộc thời gian C_1=1; C_2=2; C_3=2; C_4=3. Khi đó, ứng viên <(0,a),(3,c)> bị loại do không thỏa mãn tính chất C_2 = 2. Vì vậy:

Q_2*<0,a> = <0,ab>, <(0,a),(2,a)>*

Kiểm tra theo Định nghĩa 6 với các ứng viên trong Q_2<0,a>, khi đó ta có các mẫu dãy thường xuyên trọng số chuẩn hóa với khoảng cách thời gian có độ dài 2 với tiền tố <0,a> được nạp vào L.

Kiểm tra độ hỗ trợ với trọng số chuẩn hóa của các ứng viên <0,ab>, <(0,a),(2,a)>:

NWsupport(<0,ab>) = 2*(0,9+0,75)/2=1,65

NWsupport(<(0,a),(2,a)) = 2*(0,9+0,9)/2=1,8

Kết quả L là :

L = $L_\cup$ {<0,ab>, <(0,a),(2,a)>}

Theo tính chất đệ quy, các ứng viên và mẫu dãy thường xuyên trọng số chuẩn hóa với khoảng cách thời gian với tiền tố <0,a> sẽ tiếp tục được chia thành 2 vùng tương ứng với 2 tiền tố gồm:

- Mẫu dãy với tiền tố <0,ab >
- Mẫu dãy với tiền tố <(0,a),(2,a)>

A.1. Đối với mẫu dãy với tiền tố <0,ab>, ta xây dựng CSDL điều kiện với tiền tố <0,ab> với các mục dữ liệu trong tập ứng viên trong Q_2<0,a>.

Khi đó CSDL điều kiện với tiền tố <0,ab> sẽ bao gồm các hậu tố được xây dựng theo Định nghĩa 7:

***Bảng 4**. Cơ sở dữ liệu điều kiện với tiền tố <0,ab>*

iSID	Dãy dữ liệu
10	<(0,c),(2,ac)>

Bằng cách quét CSDL điều kiện với tiền tố <0,ab>, độ hỗ trợ của các mục dữ liệu của nó là:

support(<0,c>) = 1; support(<2,a>)= 1;
support(<2,c>)= 1

Kiểm tra theo Định nghĩa 9, tìm các mẫu ứng viên có độ dài 3 với tiền tố <0,ab>:

support(<0,c>)*MaxW = 1*0,9=0,9

support(<2,a>)*MaxW = 1*0,9=0,9

support(<2,c>)*MaxW = 1*0,9=0,9

Các ứng viên mẫu dãy thường xuyên với trọng số chuẩn hóa có độ dài 3 với tiền tố <0,ab> là:

$$Q_3<0,ab> = \varnothing$$

A.2. Đối với mẫu dãy với tiền tố <0,a),(2,a)>, ta xây dựng CSDL điều kiện với các tiền tố tương ứng với các mục dữ liệu trong tập ứng viên trong $Q_2<0,a>$. Cách khai phá mẫu dãy thường xuyên trọng số chuẩn hóa với khoảng cách thời gian với mỗi tiền tố tương ứng cũng thực hiện tương tự như Bước A.1.

B. Tìm các ứng viên và các mẫu dãy thường xuyên trọng số chuẩn hóa với khoảng cách thời gian với các tiền tố <0,b>,<0,c>

Đối với mẫu dãy với tiền tố *<0,b>,<0,c>*, ta xây dựng các CSDL điều kiện với các tiền tố tương ứng với các mục dữ liệu trong tập ứng viên trong Q_1. Cách khai phá mẫu dãy thường xuyên trọng số chuẩn hóa với khoảng cách thời gian với mỗi tiền tố tương ứng cũng thực hiện tương tự như Bước A và thực hiện để khai phá theo phương pháp đệ quy.

Bước 4: Kết quả là các mẫu dãy thường xuyên trọng số chuẩn hóa với khoảng cách thời gian trong CSDL dãy S

Các mẫu dãy thường xuyên trọng số chuẩn hóa với khoảng cách thời gian được khai phá lần lượt trong quá trình đệ quy đối với từng tiền tố. Trong phương pháp khai phá mẫu dãy thường xuyên trọng số chuẩn hóa với khoảng cách thời gian này, kết quả sẽ ít hơn các mẫu dãy thường xuyên khi không gắn thêm yếu tố trọng số của các mục dữ liệu.

III.4. Thuật toán khai phá mẫu dãy thường xuyên trọng số chuẩn hóa với khoảng cách thời gian sử dụng CSDL điều kiện theo tiền tố (WIPrefixSpan)

- Đầu vào :

(1) CSDL dãy có khoảng cách thời gian: ISBD,

(2) Ngưỡng hỗ trợ : *wminsup*,

(3) Trọng số của các mục: $w_i \in W$,

(4) Ràng buộc khoảng cách thời gian C_1, C_2, C_3, C_4,

- Đầu ra: Các mẫu dãy thường xuyên trọng số chuẩn hóa với khoảng cách thời gian L

Thuật toán WIPrefixSpan

Bắt đầu

1) Đặt $\propto = \varnothing$.

2) Đặt R $= \varnothing$; L $= \varnothing$.

3) Duyệt lần đầu ISDB, tìm các mục ứng viên có độ dài =1 và thỏa mãn điều kiện support*MaxW≥wminsup

Lặp với mọi mục i

a) Đặt $\propto =< (0, i) >$,

- Nạp R = {R, $\propto$}.

- Kiểm tra điều kiện support(∝)*NW(∝) ≥ wminsup , nếu thỏa mãn L={L, ∝}.

b) Thực hiện R = WIPrefixSpan(ISDB|∝,R,W,wminsup, C_1, C_2, C_3, C_4).

Kết thúc lặp

4) Kết quả tập L.

Kết thúc.

Thủ tục WIPrefixSpan (ISDB|∝,R,W,wminsup, C_1, C_2, C_3, C_4)

Bắt đầu:

1) Duyệt ISDB|∝ để tìm tất cả các cặp dãy với ∝ và giá trị khoảng cách thời gian giữa dãy đó với ∝ trong cặp này (định nghĩa là (Δt,i)) thỏa mãn các điều kiện support (i)* MaxW ≥ wminsup, C_1, và C_2.

2) Đặt ∝ =<∝; (Δt; i)>.

3) Kiểm tra xem có thỏa mãn điều kiện C_4

4) Chỉ khi thỏa mãn điều kiện C4

a) Thực hiện R = WIPrefixSpan (ISDB|∝,R,W,wminsup, C_1, C_2, C_3, C_4).

b) Khi ∝ thỏa mãn điều kiện C_3,

R = {R, ∝}.

c) Kiểm tra điều kiện support(∝)*NW(∝) ≥ wminsup , nếu thỏa mãn L={L, ∝}.

5) Kết quả tập L.

Kết thúc.

IV. KẾT QUẢ THỰC NGHIỆM

Trong phần này, chúng tôi trình bày kết quả thực nghiệm thuật toán WIPrefixSpan so với thuật toán do IprefixSpan [10] và thuật toán WPrefixSpan[11] trên bộ dữ liệu UCI Machine Learning: BMS-WebView với 59601 dãy dữ liệu, 497 mục dữ liệu, chiều dài trung bình 1 dãy gồm 2,42 mục dữ liệu, gồm một số dãy dài (hơn 318 dãy chứa nhiều hơn 20 mục) *(http://www.philippe-fournier-viger.com/spmf/ datasets/BMS1_spmf)*. Bộ dữ liệu BMS-WebView được sinh ngẫu nhiên các dữ liệu chiều thời gian, khoảng cách thời gian giữa 2 tập mục kề nhau trong 1 chuỗi bất kỳ được sinh ngẫu nhiên từ 0-10. Giá trị trọng số của các mục trong thuật toán WIPrefixSpan trong khoảng $0,2 \leq w_j \leq 0,9$.

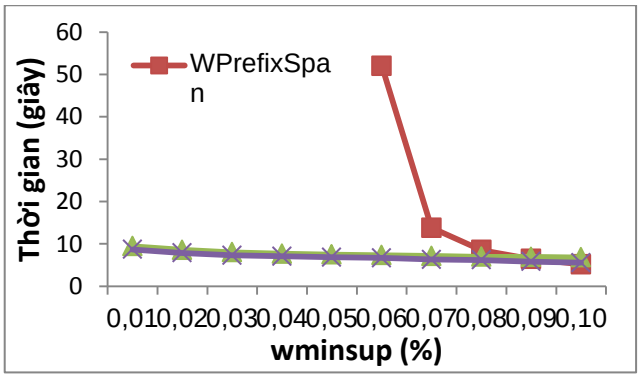
Trong phần thử nghiệm này, chúng tôi chạy thử nghiệm bộ dữ liệu BMS-WebView với các ngưỡng hỗ trợ wminsup khác nhau (từ 0,01%-0,1%). Các thuật toán IPrefixSpan[10] và WIPrefixSpanđược đưa thêm các ràng buộc thời gian: $C_1 = 0$, $C_2=3$, $C_3=0$, $C_4=50$

Tất cả các thực nghiệm được tiến hành trên một máy tính có bộ xử lý Intel Core2 Dual 2.53GHz với 3GB bộ nhớ chính, chạy Microsoft Windows XP SP3. Các thuật toán được thực hiện bởi ngôn ngữ lập trình Java 1.6 trên Eclipse.

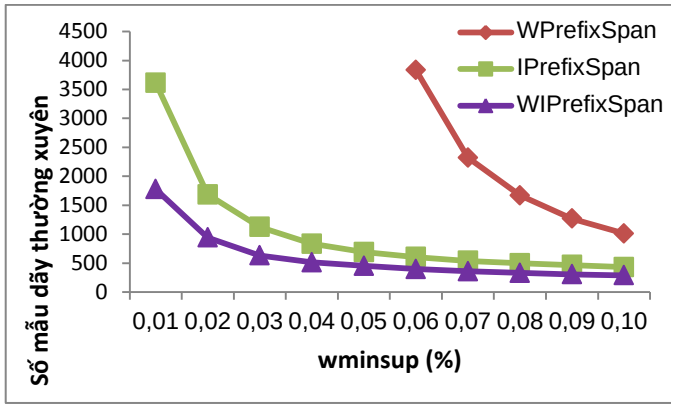
Trong trường hợp tổng quát, độ phức tạp của thuật toánWIPrefixSpan là $\mathbf{O(N^L)}$, trong đó N là số lượng các mục trong tập dữ liệu và L là chiều dài lớn nhất dãy dữ liệu trong toàn bộ các giao dịch.

- So sánh về thời gian chạy: Kết quả từ Hình 1 cho thấy khi đưa thêm ràng buộc thời gian vào thì thời gian chạy của thuật toán giảm đi đáng kể. Thuật toán WPrefixSpan có thời gian thực thi lâu hơn so với 2 thuật toán IPrefixSpan và WIPrefixSpan khi ngưỡng hỗ trợ giảm dần.
- So sánh về số mẫu dãy thường xuyên tìm được: Hình 2, ta có thể thấy thuật toán WIPrefixSpan đã giảm đáng kể số mẫu dãy thường xuyên tìm được so với 2 thuật toán IPrefixSpan và WPrefixSpan. Do thuật toán đưa thêm ràng buộc thời gian và trọng số, không gian tìm kiếm đã được rút gọn đáng kể.
- Chúng tôi thử nghiệm thuật toán WIPrefixSpan với các giá trị điều kiện từ ĐK1 đến ĐK7 với các giá trị ràng buộc thời gian C_1, C_2, C_3, C_4 tương ứng như Bảng 5,thuật toán vẫn sử dụng bộ dữ liệu BMS-WebView và bộ trọng số sinh ngẫu nhiên trong khoảng từ 0,2 đến 0,9 với

ngưỡng hỗ trợ wminsup = 0,01%. Như Bảng 5, ta có thể thấy số lượng mẫu dãy thường xuyên tìm được thay đổi theo từng điều kiện ràng buộc thời gian khác nhau. Như vậy ta có thể thông qua việc thay đổi các ràng buộc thời gian để tỉa bớt các dữ liệu không quan trọng và làm giảm không gian tìm kiếm của thuật toán.



***Hình 1**: Thời gian chạy*



***Hình 2.** Số mẫu dãy thường xuyên*

V. KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

Trong bài báo này chúng tôi nghiên cứu và phát triển thuật toán WIPrefixSpan phát hiện mẫu dãy thường xuyên trọng số chuẩn hóa với khoảng cách thời gian theo cách mô hình tăng trưởng mẫu dãy ứng viên. Với cách tiếp cận này giải thuật không cần sinh các ứng viên dãy ban đầu theo các cách tiếp cận thông thường như AprioriAll[1]. Chúng tôi sử dụng phương pháp xây dựng các CSDL điều kiện theo tiền tố cho phép thu nhỏ đáng kể không gian tìm kiếm để khai phá mẫu dãy thường xuyên. Việc đưa giá trị trọng số của các mục dữ liệu trong CSDL dãy có khoảng cách thời gian cho phép quan tâm tới sự ràng buộc giữa độ hỗ trợ, trọng số và khoảng cách thời gian của các dãy, đồng thời trong quá trình xây dựng các CSDL điều kiện theo tiền tố, sử dụng điều kiện kiểm tra để thực hiện tỉa những mục không phải là ứng viên của mẫu dãy thường xuyên trọng số chuẩn hóa với khoảng cách thời gian cho phép giảm không gian tìm kiếm nhưng vẫn đảm bảo tính phản đơn điệu trong giải thuật.

***Bảng 5**. Số mẫu dãy thường xuyên theo các giá trị điều kiện ràng buộc thời gian khác nhau*

Điều kiện	**ĐK 1**	**ĐK 2**	**ĐK 3**	**ĐK 4**	**ĐK 5**	**ĐK 6**	**ĐK 7**
Khoảng thời gian nhỏ nhất (C_1):	x	x	1	5	x	x	x
Khoảng thời gian lớn nhất (C_2):	5	5	x	x	x	x	x
Tổng quãng thời gian nhỏ nhất (C_3):	x	x	5	10	1	5	x
Tổng quãng thời gian lớn nhất (C_4):	10	50	x	x	x	x	x
Số mẫu dãy thường xuyên	966	966	1268	582	1691	1268	2109

Trong tương lai, chúng tôi sẽ tiếp tục nghiên cách thức làm giảm không gian tìm kiếm hơn nữa. Ngoài ra, chúng tôi sẽ nghiên cứu mở rộng giải thuật của chúng tôi cho bài toán khai phá mẫu chuỗi đóng.

TÀI LIỆU THAM KHẢO

[1] R.AGRAWAL, AND R.SRIKANT, *"Mining sequential patterns"*.In Proceedings of the International Conference on Data Engineering (ICDE), pp. 3-14, IEEE Computer Society (1995).

[2] R.AGRAWAL, AND R.SRIKANT, *"Mining sequential patterns: generallizations and performance improvements"*. Proceedings of the International Conference on Extending DataBase Technology (EDBT), Lecture Notes in Computer Science, Vol. 1057, pp. 3-17 (1996).

[3] J.PEI, J.HAN, B.M.ASI, H.PINO, *"PrefixSpan: Mining Sequential Patterns Efficiently by Prefix-Projected Pattern Growth"*. Proceedings of the Seventeenth International Conference on Data Engineering, pp.215-224 (2001).

[4] M.ZAKI, *"An Efficient Algorithm for Mining Frequent Sequences"*, Machine Learning, Vol. 40, pp. 31–60, 2000.

[5] J.AYRES, J.GEHRKE, T.YIU,ANDJ.FLANNICK, *"Sequential Pattern Mining using Bitmap Representation"*, in Proc. of ACM SIGKDD'02, pp. 429–435, 2002.

[6] M.S.KHAN, M. MUYEBA, F. COENEN, *"Weighted Association Rule Mining from Binary and Fuzzy Data"*. In Proceedings of 8th Industrial Conference, ICDM 2008,pp. 200-212 (2008).

[7] F.TAO, F.MURTAGH, M.FARID, *"Weighted Association Rule Mining Using Weighted Support and Significance Framework"*. In Proceedings of 9th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, pp. 661–666 (2003).

[8] U.YUN, *"An efficient mining of weighted frequent patterns with length decreasing support constraints"*, Knowledge-Based Systems, Vol. 21, No. 8, pp. 741–752 (2008).

[9] U.YUN, J.J.LEGGETT, *"WFIM: weighted frequent itemset mining with a weight range and a minimum weight"*, In 5th SIAM Int. Conf. on Data Mining, pp. 636–640 (2005).

[10] Y.HIRATE, H.YAMANA, *"Generalized Sequential Pattern Mining with Item Intervals"*, JCP,Vol. 1, No. 3, pp. 51-60 (2006).

[11] T.H.DUONG, V.D.THI, *"Thuật toán khai phá mẫu dãy thường xuyên với trọng số chuẩn hóa sử dụng CSDL tiền tố"*. Kỷ yếu hội nghị Khoa học Quốc gia lần thứ VI – Nghiên cứu cơ bản và ứng dụng CNTT (FAIR), pp. 502-511 (2013).

[12] G.C.LAN, T.P.HONG, H.Y.LEE, *"An efficient approach for finding weighted sequential patterns from sequence databases"*, Applied Intelligence, Vol. 41, No. 2, pp. 439-452 (2014).

[13] M.T.TRAN, B.LE, B.VO, *"Combination of dynamic bit vectors and transaction information for mining frequent closed sequences efficiently"*, Engineering Applications of Artificial Intelligence, Vol. 38, pp. 183-189 (2015).

[14] B.VO, F.COENEN,B.LE, *"A new method for mining Frequent Weighted Itemsets based on WIT-trees"*. Expert Systemswith Applications, Vol. 40, No. 4, pp. 1256-1264 (2013).

[15] U.YUN, G.PYUN, E.YOON, *"Efficient Mining of Robust Closed Weighted Sequential Patterns Without Information Loss"*, International Journal on Artificial Intelligence Tools, Vol. 24, No. 1, 28 pages (2015).

[16] U.YUN, K.H.RYU, *"Approximate weighted frequent pattern mining with/without noisy environments"*, Knowledge-Based Systems, Vol. 24, No. 1, pp. 73-82 (2011).

Ngày nhận bài: 11/05/2015

SƠ LƯỢC VỀ TÁC GIẢ

TRẦN HUY DƯƠNG

Sinh năm:1975

Tốt nghiệp ĐH Bách khoa Hà Nội năm 1997, ngành CNTT. Bảo vệ luận văn Thạc sĩ tại ĐH Bách khoa Hà Nội năm 2000, ngành CNTT.

Lĩnh vực nghiên cứu: Khai phá dữ liệu, cơ sở dữ liệu và học máy.

Điện thoại: 0903234934

Email: huyduong@ioit.ac.vn

VŨ ĐỨC THI

Sinh năm 1949 .

Tốt nghiệp ĐH Tổng hợp Hà Nội năm 1971. Bảo vệ luận án tiến sỹ tại Viện Hàn lâm Khoa học Hungary, năm 1987, chuyên ngành Cơ sở dữ liệu, CNTT. Nhận học hàm Phó giáo sư năm 1991, Giáo sư năm 2009.

Lĩnh vực nghiên cứu: Cơ sở dữ liệu và hệ thống thông tin, khai phá dữ liệu và học máy.

Điện thoại: 0903221304.

Email: vdthi@vnu.edu.vn