Tạp chí Khoa học Trường Đại học Cần Thơ
website: sj.ctu.edu.vn

KHAI KHOÁNG VÀ ỨNG DỤNG CỦA MẪU EPISODE MỞ RỘNG TRÊN DỮ LIỆU PHỤ THUỘC THỜI GIAN

Nguyễn Bá Diệp[1], Phạm Nguyên Khang[1], Trần Nguyễn Minh Thư[1] và Huỳnh Xuân Hiệp[2]

[1] *Bộ môn Khoa học Máy Tính, Khoa Công nghệ Thông tin & Truyền thông, Trường Đại học Cần Thơ*
[2] *Bộ môn Công nghệ Phần mềm, Khoa Công nghệ Thông tin & Truyền thông, Trường Đại học Cần Thơ*

Thông tin chung:
Ngày nhận: 03/09/2013
Ngày chấp nhận: 21/10/2013

Title:
Extended episode mining and its application in time-related data

Từ khóa:
Dữ liệu phụ thuộc thời gian, khai khoáng mẫu episode, mẫu episode mở rộng

Keywords:
Data mining in time-related data, episode mining, extended episode

ABSTRACT

In this paper, we introduce extended episode model upgraded from episode pattern in time-related data. Based on this model, we present an algorithm that finds all frequently extended episodes from an input event sequence without rescaning. By application using new characteristics of mined extended episodes, we propose an application in the diabetes data. Experimental results of this article show that the extended episodes contain useful information for prediction models.*

TÓM TẮT

*Trong bài viết này, chúng tôi giới thiệu mô hình mẫu episode mở rộng phát triển từ mô hình mẫu episode với dữ liệu phụ thuộc vào thời gian. Dựa trên mô hình vừa trình bày, chúng tôi giới thiệu giải thuật khai khoáng mẫu episode mở rộng chỉ duyệt qua chuỗi dữ liệu sự kiện 1 lần. Sử dụng những đặc tính đặc trưng của mẫu episode mở rộng chúng tôi đề xuất một ứng dụng dự đoán trên tập dữ liệu tiểu đường *. Kết quả thực nghiệm của nghiên cứu cho thấy mẫu episode mở rộng chứa nhiều thông tin hỗ trợ mô hình dự đoán.*

** dữ liệu AIM-94 cung cấp bởi Michael Kahn, Washington University, St. Louis, MO.*

1 GIỚI THIỆU

Khai khoáng dữ liệu có phụ thuộc vào thời gian là một trong mười vấn đề lớn của ngành khai khoáng dữ liệu hiện nay [15]. Việc dự đoán các sự kiện chưa xảy ra từ dữ liệu thời gian có tiềm năng ứng dụng rất lớn nhưng chưa được nghiên cứu đầy đủ. Trong các giải thuật khai khoáng mẫu tuần tự [1, 2, 3, 4, 6] đã đề ra mô hình khai khoáng *chuỗi dữ liệu con* phổ biến từ cơ sở dữ liệu. Những chuỗi dữ liệu con có mang tính chất thứ tự nhưng khó truy xuất thông tin nội tại giữa các thành tố bên trong chuỗi. Việc đánh giá kết quả dựa trên độ đo ủng hộ hay tần suất xuất hiện của chuỗi. Các giải thuật sinh ra chuỗi dữ liệu con có độ phức tạp cao, duyệt qua cơ sở dữ liệu nhiều lần nên tốn nhiều thời gian xử lý, sử dụng bộ nhớ lớn. Một phương hướng tiếp cận khác theo mô hình khai khoáng mẫu Episode [5] trình bày bởi Mannila và các đồng sự với mục tiêu tìm kiếm các episode phổ biến. Các episode này là một tập các sự kiện xuất hiện phổ biến cùng nhau trong chuỗi sự kiện. Các mẫu episode tuần tự và mẫu episode song song được sinh ra qua quá trình quét cạn toàn bộ chuỗi tuần tự. Các mô hình của tác giả đề xuất tách biệt hoàn toàn 2 loại mẫu tuần tự và mẫu song song với nhau, nhưng trong dữ liệu thực tế cho thấy 2 loại mẫu này có quan hệ mật thiết với nhau (theo mối quan hệ nguyên nhân – kết quả). Thông tin của các mẫu episode bao gồm số lần xuất hiện của mẫu cùng với mối quan hệ giữa các sự kiện trong mẫu. Tác giả có định nghĩa luật tuần tự là mối quan hệ giữa mẫu episode và mẫu *episode con,* tiêu chí để đánh giá luật tuần tự là độ tin cậy của luật và được tính

dựa trên độ ủng hộ của mẫu episode với mẫu episode con. Trong bài viết của Diệp và các đồng sự [9], tác giả đề xuất giải thuật khai khoáng mẫu episode hiệu quả hơn với lượng mẫu phổ biến nhiều hơn, thời gian xử lý nhanh và các luật tuần tự thu được có độ tin cậy cao. Giải thuật tập trung khai khoáng các mẫu episode trên những sự kiện mà người sử dụng quan tâm và các sự kiện này thật sự xuất hiện trên chuỗi dữ liệu tuần tự. Tuy lượng thông tin thu được lớn nhưng do mặt hạn chế của mô hình mẫu episode nên kết quả thu được khó ứng dụng vào trong thực tế. Trong quá trình khai khoáng mẫu episode có những episode có những thứ tự đặc biệt được Katoh gọi theo đặc trưng riêng như sectorial episode, diamond episode, elliptic episode [10, 11, 14]. Trong bài viết này chúng tôi sẽ đề xuất kết hợp mẫu tuần tự và mẫu song song episode tạo ra extended episode, việc đánh giá mẫu extended episode dựa trên độ đo ủng hộ thuần và tần suất xuất hiện thuần của mẫu.

Phần tiếp theo của bài viết này được trình bày theo thứ tự như sau: phần 2 trình bày về mô hình extended episode. Phần 3 trình bày về giải thuật khai khoáng extended episode chúng tôi đề xuất. Kết quả thực nghiệm được trình bày trong phần 4. Phần 5 là kết luận và hướng phát triển.

2 MÔ HÌNH MẪU EXTENDED EPISODE

Giả sử mỗi sự kiện xảy ra ở một thời điểm t xác định là một số tự nhiên. Cho một tập E là tập hợp các sự kiện. Một sự kiện (e, t) là một sự kiện khi và chỉ khi $e \in \mathsf{E}$ và t là một số nguyên dương chỉ thời gian xảy ra sự kiện này. Sự kiện e có thể chứa một hoặc nhiều thuộc tính nhưng để đơn giản cho việc trình bày chúng ta giả định mỗi sự kiện chỉ chứa một thuộc tính. Theo đó ta có tập sự kiện trên chuỗi sự kiện :{ (e, t) | $e \in \mathsf{E}$ } với t > 0.

Định nghĩa 1. Một chuỗi tuần tự S trên tập E là một bộ 3(S, T_s, T_e) trong đó T_s và T_e là số nguyên dương chỉ thời gian trên S. T_s là thời gian bắt đầu, T_e là thời gian kết thúc, $T_e - T_s = l_s$ và $T_s \le t_i \le T_e$ với mọi i = 1,…,n và S={ (A_1,t_1) , (A_2,t_2) , (A_3,t_3) , …. , (A_n,t_n) }là một chuỗi tuần tự các sự kiện mà $A_i \in E$ $\forall$ i = 1,…,n và $t_i \le t_{i+1}$ $\forall$ i=1,….,n-1. Định nghĩa này có một chút khác biệt với định nghĩa chuỗi tuần tự S' trong [4, 5] với S'={ (A'_1,t'_1) , (A'_2,t'_2) , (A'_3,t'_3) , …. , (A'_n,t'_n) } $A'_i \in E$ $\forall$ i = 1,…,n và $t'_i < t'_{i+1}$ $\forall$ i=1,….,n-1. Như vậy, S có thể chứa nhiều sự kiện xảy ra tại một thời điểm t so với S' chỉ có thể chứa tối đa 1 sự kiện tại một thời điểm t'.

Định nghĩa 2. Một mẫu (Episode) được xét là một bộ 3 (V, ≤ ,g) trong đó V là tập hợp của các nút, ≤ là một quan hệ thứ tự bán phần trên V và ánh xạ g: V $\rightarrow$ E là một ánh xạ kết hợp giữa các nút với các loại sự kiện. Kích thước của α kí hiệu |α| và kích thước V kí hiệu |V|. Episode α là tuần tự khi mối quan hệ ≤ là có thứ tự, Episode α là song song khi mối quan hệ ≤ là tầm thường.

Định nghĩa 3. Tần suất của mẫu X *freqX* trên chuỗi S với thời gian t_bound có công thức như sau:

$$freq(X) = \frac{\sup port(X)_{t\_bound}}{\frac{Te - Ts}{t\_bound}}$$

Trong đó freq(X) là tần suất xuất hiện của mẫu X trên chuỗi S với khoảng thời gian t_bound, support(X) $_{t\_bound}$ là độ đo ủng hộ của mẫu X trên S với khoảng thời gian t_bound.

Định nghĩa 4. Gọi σ là ngưỡng độ ủng hộ nhỏ nhất với 0 < σ < 1. Khi đó một mẫu X là mẫu phổ biến khi và chỉ khi support (X) ≥ σ.

Định nghĩa 5. Luật episode là biểu thức $\beta \Rightarrow \gamma$, với β và γ là các episodes sao cho β là episode con của γ. Episode β là episode con của γ ($\beta \prec \gamma$), nếu đồ thị biểu diễn β là đồ thị con của đồ thị biểu diễn γ. Độ tin cậy của episode γ được tính: confγ = fr(γ, S, t_bound)/ fr(β, S, t_bound)

Định nghĩa 6. Gọi λ là ngưỡng độ tin cậy nhỏ nhất với 0 < λ < 1. Khi đó luật tuần tự X là luật tin cậy khi và chỉ khi conf (X) ≥ λ.

Định nghĩa 7. Độ ủng hộ thuần của 1 mẫu episode [B|A] là độ ủng hộ episode [B|A] và không tồn tại bất kỳ episode [BX|A] ($\forall$ X ≠ B, X và B cùng xảy ra trước A) rsupport [B|A] = support [B|A] – support [B$\cap$X|A].

Định nghĩa 8. Mẫu mở rộng episode có dạng E→B, với E là mẫu episode song song, B là mẫu episode tuần tự.

3 GIẢI THUẬT KHAI KHOÁNG EXTENDED EPISODE

Việc khai khoáng các mẫu extended episode được thực hiện qua 3 giai đoạn

3.1 Giai đoạn 1 - Tạo ra các mẫu episode kèm theo các thông tin

Giải thuật IniE được sử dụng để trích xuất các mẫu episode từ chuỗi dữ liệu tuần tự đưa vào. Giải thuật được mô tả qua ngôn ngữ giả bên dưới.

Thủ tục IniE (E; t_bound; S)

Đầu vào: chuỗi tuần tự S = ⟨S1; : : : Sn⟩ với thời gian xảy ra tương ứng của mỗi sự kiện; thời gian giới hạn t_bound, Vector danh sách sự kiện E =∅, mỗi phần tử trong E chứa 1 vector lưu giá trị thời gian xuất hiện các sự kiện $E_i = \emptyset$.

Đầu ra: Vector C chứa các mẫu tìm được với thông tin thời điểm xuất hiện.

Nội dung giải thuật:

```
1  C := ∅;
2  E := ∅;
3  foreach(s ∈ S)
4     if (s ∉ E) then
5        E := E ∪ {s}
6     Es := Es ∪ {t(s)}
7  end(foreach)
8  foreach(e ∈ E)
9     foreach(α # e and α ∈ E )
10       if ([e|α] ∉ C) then
11          C:= C ∪ {[e|α]}
12       foreach(i∈Ee, j∈Eα and t(Eα[j])-t(Ee[i])
   < t_bound)
13       C[e|α] := C[e|α] ∪ [t(Ee[i]), t(Eα[j])]
14       end (foreach)
15    end (foreach)
16 end (foreach)
17 return C;
```

Giải thuật IniE duyệt qua chuỗi dữ liệu S duy nhất 1 lần, mỗi khi sự kiện xuất hiện sẽ được lưu vào vector E cùng với thời điểm xuất hiện của sự kiện. Sau khi duyệt qua hết chuỗi dữ liệu S, vector E chứa toàn bộ các sự kiện kèm thời điểm xuất hiện của các sự kiện. Lần lượt xét các sự kiện trong E, với mỗi cặp sự kiện khác nhau có thời điểm xuất hiện thỏa mãn tham số t_bound sẽ tạo mẫu tuần tự episode tương ứng với hai sự kiện khác nhau đó, thông tin về thời điểm xuất hiện cũng sẽ được lưu vào vector C. Vòng lặp kết thúc khi không tìm được thêm các cặp sự kiện thỏa mãn yêu cầu. Kết thúc giai đoạn 1 ta thu được vector C chứa danh sách các mẫu episode và thời điểm xuất hiện của từng mẫu episode, vector E chứa các sự kiện và thời điểm xuất hiện của các sự kiện.

Định lý 1: Độ phức tạp của IniE là $O(n+ l^2m)$.

Chứng minh: Với vòng lặp đầu tiên duyệt qua chuỗi tuần tự S để lưu lại thông tin và thời điểm xảy ra sự kiện có độ phức tạp tuyến tính phụ thuộc vào độ lớn n của chuỗi sự kiện S. Trong vòng lặp, mỗi thời điểm sự kiện s xuất hiện trong S sẽ được lưu thời gian lại theo loại trong vector sự kiện (E) nên thời gian tính toán sẽ tuyến tính theo **n**. Vậy độ phức tạp từ dòng 1-7 là O (n). Vòng lặp thứ 2 từ dòng 8 đến dòng 16 lặp qua từng loại sự kiện trong vector sự kiện E. Vòng lặp này chứa vòng lặp con lặp lại qua từng sự kiện trong E và không xét 2 sự kiện trùng nhau nên độ phức tạp lúc này là l^2 (với l là số sự kiện trong E). Với mỗi mẫu tuần tự episode được tạo ra, lần lượt xét thời gian từng cặp sự kiện tạo nên mẫu với vòng lặp từ dòng 11 đến dòng 14. Vòng lặp này sẽ lặp qua **m** lần thời gian của sự kiện xuất hiện nhiều (α hoặc e). Mỗi cặp thời gian thỏa ràng buộc tổng thời gian nhỏ hơn độ lớn t_bound sẽ được thêm vào vector C tương ứng với mẫu được tạo ra bởi 2 sự kiện đó. Như vậy độ phức tạp thủ tục IniE là $O (n + l^2m)$. Trong thực tế, số lần xuất hiện **m** của các sự kiện và số sự kiện l trong E nhỏ hơn độ lớn **n** của chuỗi dữ liệu S nhiều lần nên giải thuật IniE là chấp nhận được.

3.2 Giai đoạn 2 – Tạo các mẫu Episode dạng mở rộng và hiệu chỉnh thông tin trên các mẫu

Giải thuật ExtEE đọc thông tin các mẫu episode trong vector C và thông tin các sự kiện trong vector E tạo ra ở giai đoạn 1. Đầu tiên khởi tạo vector M để chứa các mẫu mở rộng và thông tin độ đo ủng hộ thuần. Lần lượt xét các cặp mẫu trong vector C, với mỗi cặp mẫu có cùng sự kiện cuối lần lượt xét thời điểm xuất hiện của cặp mẫu. Nếu mẫu mở rộng chưa có trong vector M thì thêm vào vector M. Mỗi cặp thời điểm thỏa giá trị t_bound sẽ được lưu lại trong vector M theo tương ứng với từng mẫu trong M và cập nhật độ đo ủng hộ của các mẫu trong C. Các cặp thời điểm thỏa giá trị t_bound sẽ có 3 trường hợp xảy ra: 2 mẫu cùng xuất hiện đúng vị trí trong thời gian t_bound, mẫu thứ nhất có thời điểm bắt đầu và thời điểm kết thúc nhỏ hơn thời điểm mở đầu và kết thúc của mẫu thứ hai (hoặc ngược lại), mẫu thứ nhất được chứa trong mẫu thứ hai (hoặc ngược lại). Kết thúc giải thuật, vector M sẽ chứa các mẫu mở rộng với độ đo ủng hộ thuần. Giải thuật được mô tả chi tiết bằng ngôn ngữ giả như sau:

Thủ tục ExtEE (E; C; t_bound)
Đầu vào: Vector C chứa các mẫu Episode tuần tự, Vector E chứa danh sách sự kiện thu được ở giai đoạn 1, thời gian giới hạn t_bound,
Đầu ra: Vector M chứa các mẫu mở rộng với độ ủng hộ thuần của từng mẫu;
Giải thuật:

```
1  M := ∅;
2  foreach(A[α|x] ∈ C, B[β|x] ∈ C and α ≠ β )
3     M:= M ∪ {[αβ|x]}
4     foreach([i1,i2]∈A[α|x], [j1,j2]∈Eβ)
5        if ([i1,i2]⊂ [j1,j2]) then
6           M[αβ|x] := M[αβ|x] ∪ [j1,j2]
7           C[α|x]:= C[α|x] – [i1,i2]
8           C[β|x]:= C[β|x] – [j1,j2]
9        if([i1,i2]⊃ [j1,j2]) then
10          M[αβ|x] := M[αβ|x] ∪ [i1,i2]
11          C[α|x]:= C[α|x] – [i1,i2]
12          C[β|x]:= C[β|x] – [j1,j2]
13    end(foreach)
14    if(M[αβ|x] = ∅)
15       M:= M - {[αβ|x]}
16 end(foreach)
17 return M;
```

Định lý: Độ phức tạp của ExtEE là O (n.m)

Chứng minh: ExtEE thực hiện chủ yếu với vòng lặp từ dòng 2 đến dòng 16. Vòng lặp này sẽ lặp qua **n** cặp mẫu tuần tự episode có chứa cùng 1 loại sự kiện X ở thứ tự cuối cùng của mẫu trong vector chứa mẫu C (vector mẫu tạo được ở giai đoạn 1). Vòng lặp này chứa một vòng lặp con từ dòng 5 đến dòng 13, vòng lặp bên trong này sẽ lặp qua **m** các cặp thời gian của mẫu episode A và B. Khi các cặp thời gian này thỏa mẫu A và B cùng xuất hiện trong khung thời gian và khung thời gian này nhỏ hơn ngưỡng t_bound thì sẽ tạo mẫu mở rộng có dạng [(AB) →X]. Nếu M không chứa mẫu dạng này thì sẽ thêm vào M. Nếu M đã chứa mẫu thì thêm khoảng thời gian lớn nhất chứa cả 2 mẫu tuần tự. Độ phức tạp của 2 vòng lặp này là O (n.m).

3.3 Giai đoạn 3 – Kết hợp thông tin để dự đoán

Sử dụng vector M chứa các mẫu mở rộng cùng với thông tin độ đo ủng hộ thuần thu được từ giai đoạn 2 để sinh ra các luật tuần tự thuần. Việc dự đoán chủ yếu dựa trên luật tuần tự thuần với công thức như sau:

$$p(Y) = \sum [X \mid Y]_{rconf} \ (\forall\, T_S\text{-}[X|Y]_{ts} \leq t\_bound)$$

Trong đó T_S là thời điểm kết thúc của chuỗi dữ liệu sự kiện, ts là thời điểm xuất hiện cuối cùng của mẫu có dạng [X|Y] với Y là một sự kiện bất kỳ (Y∈E), rconf là độ tin cậy thuần của mẫu [X|Y].

4 KẾT QUẢ THỰC NGHIỆM

Áp dụng các giải thuật khai khoáng mẫu extended episode vào cơ sở dữ liệu tiểu đường AIM-94 cung cấp bởi Michael Kahn, Washington University, St. Louis, MO. Dữ liệu bao gồm 70 chuỗi dữ liệu tuần tự (tương ứng với 70 bệnh nhân). Mỗi chuỗi trung bình có 500 mốc thời gian với 700 sự kiện xảy ra. Dữ liệu có 20 loại sự kiện bao gồm 12 loại là dữ liệu liên tục và 8 loại là dữ liệu rời rạc. Trong 12 loại dữ liệu liên tục có 3 loại là các liều insulin và 9 loại còn lại là chỉ số đường trong máu. Đơn vị thời gian được tính là giờ, khoảng thời gian đo thông thường là các giờ: ăn sáng (08:00), ăn trưa (12:00), ăn chiều (18:00) và ăn khuya (22:00). Sau khi chuẩn hóa dữ liệu chúng tôi thu được các loại sự kiện sau:

Có 9 loại sự kiện về lượng đường trong máu được minh họa chi tiết trong Bảng 1:

Bảng 1: Bảng rời rạc dữ liệu đường trong máu

Chỉ số đường trong máu	Sự kiện
Tăng trên 25%	Tăng rất mạnh
Tăng 17% → 25%	Tăng mạnh
Tăng 12% → 17%	Tăng vừa
Tăng 5% → 12 %	Tăng nhẹ
Dao động 5% → 5%	Không đổi
Giảm 5% → 12%	Giảm nhẹ
Giảm 12 → 17%	Giảm vừa
Giảm 17 → 25%	Giảm mạnh
Giảm hơn 25%	Giảm rất mạnh

Có 9 loại sự kiện về liều lượng insulin được minh họa trên các Bảng 2, 3, 4.

Bảng 2: Bảng rời rạc dữ liệu Regular insulin

Lượng insulin tác dụng nhanh (Regular insulin dose)	Sự kiện
Dưới 6 đơn vị	Liều R1
Từ 6 đến 12 đơn vị	Liều R2
Trên 12 đơn vị	Liều R3

Bảng 3: Bảng rời rạc dữ liệu NHP insulin

Lượng insulin tác dụng trung gian (NHP insulin dose)	Sự kiện
Dưới 12 đơn vị	Liều NHP1
Từ 12 đến 20 đơn vị	Liều NHP2
Trên 20 đơn vị	Liều NHP3

Bảng 4: Bảng rời rạc dữ liệu UltraLente insulin

Lượng insulin tác dụng chậm (UltraLente insulin dose)	Sự kiện
Dưới 7 đơn vị	Liều U1
Từ 7 đến 13 đơn vị	Liều U2
Trên 13 đơn vị	Liều U3

Ngoài ra theo quan điểm y học, chúng tôi chọn từ cơ sở dữ liệu 13 sự kiện sau: trước khi ăn sáng, sau khi ăn sáng, trước khi ăn trưa, sau khi ăn trưa, trước khi ăn vặt, có triệu chứng hạ đường trong máu, bữa ăn kiêng thông thường, bữa ăn kiêng nhiều hơn thông thường, bữa ăn kiêng ít hơn thông thường, hoạt động thể dục thông thường, hoạt động thể dục nhiều hơn thông thường, hoạt động thể dục ít hơn thông thường, sự kiện đặc biệt (tiệc, lễ hội...). Tổng hợp lại có 31 loại sự kiện.

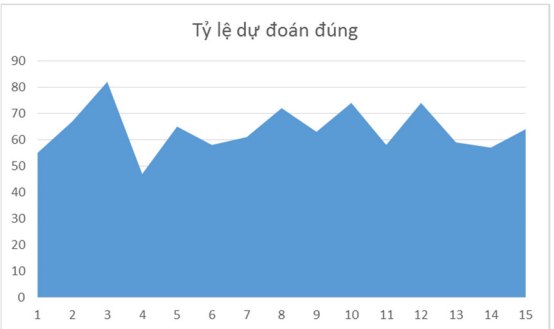
Các thông số cho chương trình được xác định: ràng buộc thời gian t_bound = 24 (giờ), ngưỡng ủng hộ nhỏ nhất $support_{min}$ = 0,2 %, ngưỡng ủng hộ thuần $rsupport_{min}$ = 0,1%. Giải thuật được cài đặt trên hệ điều hành Window, viết bằng ngôn ngữ lập trình JAVA với công cụ phát triển NETBEAN 7.0.

Bảng 5: Kết quả áp dụng các giải thuật và độ đo khác nhau trên tập dữ liệu AIM-94

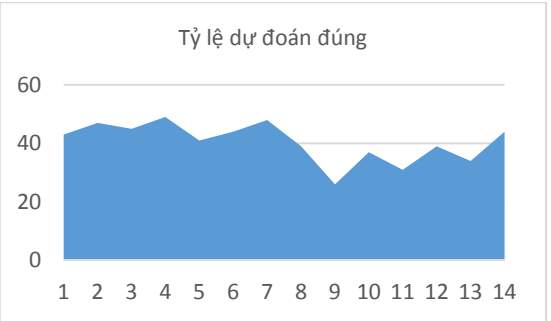
Độ đo	Episode	Frequent Episode	Extended Episode	Frequent Extended Episode
support	823	672	1356	1219
rsupport	821	631	1352	936

Bảng 6: Thời gian thực thi của các giải thuật trên các chuỗi dữ liệu khác nhau

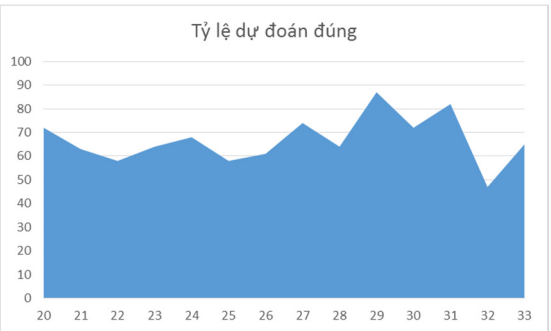
Thời gian thực hiện (ms)	Data-01	Data-12	Data-39	Thời gian Trung bình
WINEPI	823	672	645	713
MINEPI	730	631	589	650
DYNEPI	525	487	471	494
IniE + ExtEE	780	656	612	682



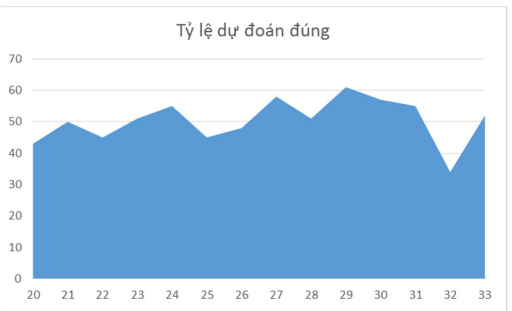
Hình 1: Trung bình kết quả dự đoán trên từng chuỗi dữ liệu sử dụng độ đo ủng hộ thuần, mẫu extended episode và luật tuần tự thuần (đơn vị:%)



Hình 2: Trung bình kết quả dự đoán trên từng chuỗi dữ liệu sử dụng độ đo ủng hộ, mẫu episode và luật tuần tự (đơn vị:%)



Hình 3: Trung bình kết quả dự đoán trên từng chuỗi dữ liệu sử dụng độ đo ủng hộ thuần, mẫu extended episode và luật tuần tự thuần (đơn vị:%)



Hình 4: Trung bình kết quả dự đoán trên từng chuỗi dữ liệu sử dụng độ đo ủng hộ, mẫu extended episode và luật tuần tự (đơn vị:%)

Bảng so sánh ở Bảng 5 cho thấy, khi sử dụng độ đo ủng hộ thuần (rsupport), tổng số mẫu episode và extended episode luôn thấp hơn khi sử dụng độ đo ủng hộ (support). Vì độ đo ủng hộ thuần xác định chính xác số lần mẫu xuất hiện trong khoảng thời gian t_bound giúp tránh trường hợp 1 mẫu được xét ủng hộ 2 lần. Thông tin dự đoán chủ yếu dựa trên luật tuần tự thuần, mà luật tuần tự thuần phụ thuộc chủ yếu vào độ ủng hộ thuần của mẫu. Do đó kết quả dự đoán sử dụng độ đo ủng hộ thuần

và luật tuần tự thuần cho kết quả chính xác hơn (Hình 3 và Hình 4).

Hình 2 cho thấy việc sử dụng mẫu episode để dự đoán cho kết quả có độ chính xác thấp nhất trong cả 4 trường hợp.

Bảng 6 cho thấy tổng thời gian thực hiện của giải thuật cả 2 giải thuật IniE và ExtEE tốt hơn giải thuật WINEPI.

5 KẾT LUẬN VÀ ĐỀ XUẤT

Chúng tôi vừa trình bày mô hình mẫu episode mở rộng (extended episode) thay thế cho mẫu episode trong các ứng dụng dự đoán dựa trên dữ liệu thời gian. Trong thực tế với trường hợp dữ liệu đa sự kiện (các sự kiện xảy ra cùng một thời điểm) thì không thể sử dụng mẫu episode được. Ngoài ra chúng tôi còn đề xuất độ đo ủng hộ thuần, độ tin cậy thuần thay thế cho độ đo ủng hộ và độ tin cậy. Kết quả thực nghiệm cho thấy các cải tiến trên hỗ trợ tốt hơn cho việc dự đoán sự kiện tương lai. Trong thời gian tới, nhóm chúng tôi sẽ cải tiến mô hình hoạt động linh hoạt hơn có thể tận dụng được thông tin dữ liệu mẫu episode có sẵn và có thể phát hiện các sự kiện *ẩn* trong chuỗi dữ liệu.

TÀI LIỆU THAM KHẢO

1. R. Agrawal, H. Mannila, R. Srikant, H. Toivonen, A. I. Verkamo: Fast discovery of association rules, in U. M. Fayyed, G. Piatetsky-Shapiro, P. Smyth, R. Uthurusamy (eds.): Advances in Knowledge Discovery and Data Mining, AAAI/MIT Press, 307–328, 1996.
2. R. Agrawal, R. Srikant: Fast algorithms for mining association rules in large databases, Proc. 20th VLDB, 487–499, 1994.
3. R. Agrawal, R. Srikant: Mining sequential patterns, Proc. 11th ICDE, 3–14, 1995.
4. C. Bettini, S. Wang, S. Jajodia, J.-L. Lin: Discovering frequent event patterns with multiple granularities in time sequences, IEEE Trans. Knowledge and Data Engineering10, 222–237, 1998.
5. H. Mannila, H. Toivonen, A. I. Verkamo: Discovery of frequent episodes in event sequences, Data Mining and Knowledge Discovery 1, 259–289, 1997.
6. J. Pei, J. Han, B. Mortazavi-Asi, J. Wang, H. Pinto, Q. Chen, U. Dayal, M.-C. Hsu: Mining sequential patterns by pattern-growth: The PrefixSpan approach, IEEE Trans. Knowledge and Data Engineering16, 1–17, 2004.
7. R. Srikant, R. Agrawal: Mining sequential patterns: Generalizations and perfor-mance improvements, Proc. 5th EDBT, 3–17, 1996..
8. S. Tsumoto: Guide to the bacteriological examination data set, in E. Suzuki (ed.): Proc. International Workshop of KDD Challenge on Real-World Data (KDD Challenge 2000), 8–12, 2000.
9. Ba-Diep Nguyen, Xuan-Hiep Huynh, Julien Blanchard : Phát hiện mẫu tuần tự với kích thước thay đổi bằng giải thuật DYNEPI, Kỷ yếu Hội nghị khoa học 5 năm nghiên cứu khoa học Khoa CNTT Trường Đại học Cần Thơ, 2011.
10. Katoh, T., Hirata, K., Harao, M.: Mining frequent diamond episodes from event sequences. In: Torra, V., Narukawa, Y., Yoshida, Y. (eds.) MDAI 2007. LNCS (LNAI), vol. 4617, pp. 477–488. Springer, Heidelberg (2007).
11. Katoh, T., Hirata, K., Harao, M., Yokoyama, S., Matsuoka, K.: Extraction of sectorial episodes representing changes for drug resistant and replacements of bacteria. In: Proc. CME 2007, pp. 304–309 (2007).
12. Katoh, T., Arimura, H., Hirata, K.: A polynomial-delay polynomial-space algorithm for extracting frequent diamond episodes from event sequences. In: Theeramunkong, T., Kijsirikul, B., Cercone, N., Ho, T.-B. (eds.) PAKDD 2009. LNCS (LNAI), vol. 5476, pp. 172–183. Springer, Heidelberg (2009).
13. Katoh, T., Hirata, K.: A simple characterization on serially constructible episodes. In: Washio, T., Suzuki, E., Ting, K.M., Inokuchi, A. (eds.) PAKDD 2008. LNCS (LNAI), vol. 5012, pp. 600–607. Springer, Heidelberg (2008).
14. Katoh, T., Hirata, K.: Mining frequent elliptic episodes from event sequences. In: Proc. 5th LLLL, pp. 46–52 (2007).
15. 1Q. Yang and X. Wu. 10 Challenging Problems in Data Mining Research. Journal of Information Technology & Decision Making 5(4):597-604, 2006.