PHƯƠNG PHÁP ẨN LUẬT KẾT HỢP DỰA TRÊN TIẾP CẬN GIÀN GIAO

Lê Quốc Hải
Trường Cao đẳng Sư phạm Quảng Trị

TÓM TẮT

Ẩn các luật kết hợp nhạy cảm là bài toán quan trọng trong khai phá các luật kết hợp. Một trong những vấn đề đặt ra khi giải quyết bài toán này là giảm các hiệu ứng phụ, tức là giảm các luật bị ẩn nhầm và các luật mới được sinh ra, và giảm số lần truy cập dữ liệu. Bài báo giới thiệu một hướng tiếp cận mới dựa trên lý thuyết giàn giao. Thuật toán HidingRules thu được là có cơ sở toán học chặt chẽ, sử dụng heuristic để xác định các mục, các giao tác cần phải sửa đổi nhằm ẩn các luật kết hợp nhạy cảm sao cho hiệu ứng phụ là thấp nhất.

1. Đặt vấn đề

Khai phá dữ liệu là một lĩnh vực nghiên cứu khá mới của ngành khoa học máy tính. Các nghiên cứu gần đây chủ yếu tập trung vào việc phát triển các thuật toán phục vụ cho quá trình phân tích dữ liệu từ kho dữ liệu. Phân tích các *luật kết hợp* là một trong những phương pháp của khai phá dữ liệu. Nhiệm vụ của phương pháp này là phân tích dữ liệu trong cơ sở dữ liệu nhằm phát hiện và đưa ra những mối liên hệ về giá trị dữ liệu. Đó chính là các tập luật kết hợp. Thông thường, các luật kết hợp được khai thác từ các *bảng giao tác*, mỗi bảng giao tác được xác định gồm các mục (cột) và các giao tác (dòng). Hợp của các mục gọi là *tập mục*, chẳng hạn *XY*. Mỗi luật kết hợp thu được từ bảng giao tác là quan hệ hai ngôi giữa hai tập mục *X* và *Y*, ký hiệu *X*=>*Y*, được sinh ra từ các tập mục thường xuyên *XY* có tần suất xuất hiện trên một ngưỡng hỗ trợ tối thiểu δ nào đó. Trong khai phá các luật kết hợp, người ta chỉ quan tâm đến các luật có độ hỗ trợ lớn hơn hoặc bằng một *ngưỡng hỗ trợ tối thiểu* (*minsup*) và độ tin cậy lớn hơn hoặc bằng một *ngưỡng tin cậy tối thiểu cho trước* (*minconf*) gọi là các luật kết hợp *phổ biến*. Một vấn đề thường gặp là khi cung cấp dữ liệu cho các trung tâm khai thác tri thức, một số cơ sở không muốn công bố các luật vi phạm đến tính riêng tư của cá nhân hoặc của xí nghiệp. Thí dụ, nếu *X* là tập mục về thương hiệu *xe máy Honda*, *Y* là tập mục về *số vụ tai nạn xe máy* thì việc công bố tương quan giữa *X* và *Y* sẽ mang đến sự bất lợi cho việc kinh doanh xe máy Honda. Các luật *X*=>*Y* như trên được gọi là các luật kết hợp nhạy cảm. Vì thế, các cơ sở cung cấp dữ liệu sẽ phải loại bỏ các luật kết hợp nhạy cảm *X*=>*Y* sao cho chúng không thể được khai thác bởi các thuật toán khai phá dữ liệu. Việc loại bỏ (*ẩn*) này được thực hiện bằng cách sửa bảng giao tác sao cho độ hỗ trợ của luật hoặc độ tin cậy của luật giảm xuống dưới ngưỡng nào đó. Hướng nghiên cứu này là rất cần

thiết khi muốn bảo vệ bí mật riêng tư trong khai phá dữ liệu.

Bài báo này đề xuất một tiếp cận mới cho bài toán ẩn các luật kết hợp nhạy cảm. Vận dụng lý thuyết giàn giao ta có thể xác định một cận trên đúng và sau đó là cận dưới đúng đối với một tập mục cho trước, xem xét các tập mục này trong các luật kết hợp chứa nó để ẩn luật là mục tiêu của tiếp cận này. Hướng tiếp cận này có những điểm mới sau đây. Thứ nhất, lần đầu tiên sử dụng lý thuyết giàn giao vào bài toán ẩn luật kết hợp. Thứ hai, nhờ vận dụng các tính chất của giàn giao đã chỉ ra rằng có thể vận dụng lý thuyết đồ thị để xác định các *tập mục gây ảnh hưởng* và các *tập mục chịu ảnh hưởng* trực tiếp khi sửa giao tác trên tập mục thuộc luật nhạy cảm do đó làm giảm thời gian truy xuất các giao tác và không gây ra các hiệu ứng phụ theo nghĩa là ẩn nhầm các luật kết hợp không nhạy cảm hoặc sinh ra các luật kết hợp mới.

2. Phát biểu bài toán

Cho một bảng trị T 0/1 gồm N dòng và M cột. Các cột được gán tên lần lượt $A, B, C,\ldots$ lấy từ một tập hữu hạn các phần tử U.

Mỗi phần tử trong U gọi là một *mục,* mỗi tập con X của U gọi là một *tập mục*. Mỗi dòng t của bảng T được gọi là một *giao tác.* Ta ký hiệu tập mục như một dãy các kí tự viết liền nhau, hợp của hai tập mục X và Y được kí hiệu là XY. Với mỗi giao tác $t \in T$ và mỗi mục $A \in U$ ta kí hiệu $t.A$ là giá trị tương ứng xuất hiện trên giao của giao tác t và cột A trong bảng T. Như vậy $t.A \in \{0,1\}$. Ta định nghĩa $Set(t)$ là tập mục tại đó t nhận trị 1, $Set(t) = \{A \in U \mid t.A = 1\}$. Nếu $X \subseteq Set(t)$ thì ta nói *giao tác t chứa tập mục X*. Với mỗi tập mục $X \subseteq U$ ta xác định $\alpha(X)$ là *số lượng giao tác* chứa X, $\alpha(X) = \|\{t \in T \mid X \subseteq Set(t)\}\|$, trong đó, kí hiệu $\|M\|$ cho biết *lực lượng* (*số phần tử*) của tập M. Tỷ số $\alpha(X)/N$ được gọi là *độ hỗ trợ* của tập mục X. Với N cho trước và cố định, ta có thể coi $\alpha(X)$ là độ hỗ trợ của tập mục X. Cho trước giá trị σ và gọi là *ngưỡng hỗ trợ tối thiểu.* Các tập mục X thỏa tính chất $\alpha(X) > \sigma$ được gọi là các *tập mục thường xuyên*. Từ tập mục thường xuyên M có thể sinh ra các luật kết hợp thể hiện mối liên hệ giữa các tập mục con của M, một luật $X=>Y$ có thể được sinh ra từ M nếu chúng thỏa X, $Y \subset M$, $X \cap Y = \varnothing$ và $X \cup Y = M$. Trong bài toán khai thác các luật kết hợp, người ta chỉ xem xét các luật kết hợp có giá trị, được biểu thị thông qua độ hỗ trợ và độ tin cậy của luật. *Độ hỗ trợ* của luật X=>Y được xác định là $\alpha(X=>Y) = \alpha(X \cup Y) = \|\{t \in T \mid XY \subseteq Set(t)\}\|$. *Độ tin cậy* của luật $X=>Y$ được xác định là $\beta(X=>Y) = \alpha(X \cup Y)/\alpha(X)$. Các luật kết hợp được coi là có giá trị khi độ hỗ trợ của nó nằm trên *ngưỡng hỗ trợ tối thiểu* δ và độ tin cậy nằm trên *ngưỡng tin cậy tối thiểu* σ nào đó, các luật như vậy gọi là *luật kết hợp phổ biến*.

Một luật kết hợp phổ biến được gọi là được *ẩn* nếu ta giảm độ hỗ trợ của nó xuống dưới ngưỡng δ hoặc giảm độ tin cậy của nó xuống dưới ngưỡng σ, nghĩa là ta không thể khai thác nó trong bảng giao tác bằng các kỹ thuật khai thác luật kết hợp.

Bài toán **ẩn các luật kết hợp** được phát biểu như sau:

Cho một bảng giao tác T, một tập luật kết hợp R được khai thác từ T và một tập luật nhạy cảm $R_S \in R$, làm thế nào có thể chuyển đổi bảng T thành bảng T' sao cho các luật trong R vẫn có thể khai thác được, ngoại trừ các luật trong R_S.

Ví dụ 1: Cho bảng giao tác a), với ngưỡng hỗ trợ tối thiểu δ = 4, các tập mục thường xuyên b) và với ngưỡng tin cậy tối thiểu σ = 70% thì các luật kết hợp được sinh ra trong bảng c)

a) Bảng giao tác

Số hiệu	*ABCDE*
1	1 1 1 1 1
2	1 1 1 0 1
3	1 1 0 1 1
4	0 1 1 1 1
5	0 1 0 1 1
6	0 0 1 1 1
7	0 0 0 1 1

b) Tập mục thường xuyên

Tập mục thường xuyên	Độ hỗ trợ
B	5
C	4
D	6
E	7
BD	4
BE	5
CE	4
DE	5
BDE	4

c) Luật kết hợp phổ biến

Luật kết hợp phổ biến	β	α
B=>D	80%	4
B=>E	100%	5
D=>E	82%	5
E=>D	70%	5
E=>B	70%	5
B=>DE	80%	4
BD=>E	100%	4
BE=>D	80%	4
DE=>B	80%	4

Để ẩn một luật, chẳng hạn *E=>B*, có hai tiếp cận: thứ nhất là giảm độ hỗ trợ của tập mục sinh ra luật là *BE* xuống dưới ngưỡng hỗ trợ tối thiểu, chẳng hạn sửa B trong giao tác có số hiệu 1 và 2 từ giá trị 1 thành giá trị 0, khi đó $\alpha(BE) = 3 < \delta$ nên luật *E=>B* không thể được sinh ra, tuy nhiên, trong tình huống này thì các tập mục *B*, *BE*, *BD*, *BDE* cũng bị ẩn đi và do đó các luật kết hợp được sinh ra từ đó cũng bị ẩn đi là *E=>B*, *BE=>D*, *DE=>B*, *BD=>E*, *B=>DE*. Tiếp cận thứ hai là giảm độ tin cậy của luật xuống dưới ngưỡng σ. Chẳng hạn, ở đây sửa mục *B* trên một giao tác chứa *BE* có số hiệu 1. Khi đó $\alpha(BE)=4>\delta$ nhưng $\beta(E=>B)=58\% < \sigma$ nên luật *B=>E* được ẩn. Vấn đề đặt ra cho bài toán này là cần phải lựa chọn các mục và các giao tác để sửa đổi giá trị sao cho *hiệu ứng phụ* là nhỏ nhất, đó là số các luật bị ẩn nhầm và số các luật mới được sinh ra và số lần truy cập dữ liệu là ít nhất.

Tổng quát, để ẩn luật phổ biến *X=>Y* ta có thể dựa trên hai tiếp cận là:

- Giảm độ hỗ trợ của tập mục sinh luật *XY* xuống dưới ngưỡng hỗ trợ tối thiểu δ.

- Giảm độ tin cậy của luật xuống dưới ngưỡng tin cậy tối thiểu σ.

Trong tiếp cận thứ nhất, rõ ràng số luật bị ẩn nhầm có thể sẽ nhiều bởi vì khi tập mục *M* bị ẩn thì tất cả các luật sinh ra từ tập *M* đều bị ẩn đồng thời tất cả các luật được sinh ra từ các tập mục thường xuyên chứa *M* cũng bị ẩn theo. Tiếp cận thứ hai giảm được các luật bị ẩn nhầm. Đối với luật *X=>Y*, độ tin cậy của luật được xác định là $\beta(X=>Y) = \alpha(XY)/\alpha(X)$. Nếu giảm β bằng cách sửa *X* trên các giao tác chứa *XY* thì cả tử số và mẫu số đều giảm, như thế sẽ làm cho tốc độ hội tụ của thuật toán ẩn luật bị chậm. Do đó, ta giảm β bằng cách sửa *Y* trên các giao tác chứa *XY*, khi đó chỉ có tử số là $\alpha(XY)$ bị giảm mà mẫu số $\alpha(X)$ không thay đổi, và do đó, tốc độ hội tụ của thuật toán là nhanh

hơn. Như vậy, ta cần phải lựa chọn mục A trong Y và sửa A từ *1* thành *0* nhằm ẩn luật $X=>Y$ sao cho hiệu ứng phụ là thấp nhất. Mục tiếp theo trình bày lý thuyết giàn giao và cơ sở vận dụng vào bài toán ẩn luật kết hợp.

3. Lý thuyết giàn giao

Định nghĩa 3.1. Tập hợp được sắp thứ tự V được gọi là một giàn nếu với hai phần tử bất kì a,b ∈ V, tập hợp {a,b} luôn có cận trên và cận dưới. Cận trên và cận dưới của {a,b} được kí hiệu lần lượt là a ∨ b và a ∧ b.

Mệnh đề 3.1. *Một họ các tập con G trên tập hữu hạn U với phép toán bao hàm (⊆) tạo thành một giàn.*

Cho tập hữu hạn U gọi là *tập nền*, ta kí hiệu $Poset(U)$ *là họ toàn thể các tập con của U* với thứ tự bộ phận là phép *bao hàm* $\subseteq$, $Poset'(U) = Poset(U) - \{U\}$. *Một* giàn giao G *là một họ các tập con của U đóng với phép giao,* cụ thể là*,* nếu $G = \{V_1, V_2, ..., V_k \mid V_i \in Poset(U), i = 1,2,\ldots,k\}$ thì $\forall\ V_i, V_j \in G$: $V_i \cap V_j \in G$. Khi đó G chứa *duy nhất* một họ con S sao cho mọi phần tử của G đều được biểu diễn qua giao của các phần tử trong S*,* cụ thể là*,* S là tập con nhỏ nhất của G thỏa tính chất $G = \{Y \mid Y = X_1 \cap \ldots \cap X_k,\ k \geq 0, X_1, \ldots, X_k \in S\}$. S được gọi là *tập sinh của giàn G* và được ký hiệu là $Gen(G)$. Theo quy ước, giao của một họ rỗng các tập con chính là U*,* do đó mọi Gen đều không chứa U. Trong [1] trình bày và chứng minh tính đúng của thuật toán tìm tập sinh của giàn giao G cho trước.

Cho $(M, \leq)$ là một tập hữu hạn có thứ tự bộ phận. Phần tử m trong M được gọi là *cực đại* nếu từ $m \leq x$ và $x \in M$ ta luôn có $m=x$. Ta ký hiệu $MAX(M)$ là tập các phần tử cực đại của M. Dễ thấy rằng, với mỗi phần tử x trong M*,* luôn tồn tại một phần tử m trong $MAX(M)$ thỏa $x \leq m$. Với mỗi họ các tập con của một tập hữu hạn U cho trước ta xét thứ tự bộ phận $\subseteq$. Cho G là một giàn giao trên tập hữu hạn U. Ta ký hiệu $Coatom(G) = MAX(G - \{U\})$ và gọi các phần tử trong $Coatom(G)$ là *đối nguyên tử* của giàn giao G. Trong [1] đã đưa ra thuật toán Gen(G) tìm Gen của một tập hữu hạn G.

Mệnh đề 3.2.

Cho $(M,\leq)$ *là một tập hữu hạn có thứ tự bộ phận và* $P \subseteq Q \subseteq M$*. Khi đó, nếu* $x \in MAX(Q)$ và $x \in P$ *thì* $x \in MAX(P)$.

Chứng minh

Theo giả thuyết, x ∈ MAX(Q) suy ra ∀ m ∈ Q, nếu x ≤ m thì x = m (1). Do x ∈ P và theo (1) với m thỏa x ≤ m => x = m nên m ∈ P. Theo định nghĩa suy ra x ∈ MAX(P) ■

Bổ đề 3.1.

Với mọi giàn giao G trên tập hữu hạn U ta có MAX(Gen(G)) = MAX(G\{U})

Chứng minh:

Giả sử $X \in MAX(Gen(G))$. Khi đó, do $X \neq U$ nên $X \in G \setminus \{U\}$. Nếu $Y \in G \setminus \{U\}$ và $X \subseteq Y$ thì theo định nghĩa của tập sinh, Y được biểu diễn qua một giao của các phần tử trong $Gen(G)$,

$$Y = Z_1 \cap Z_2 \cap ... \cap Z_k \; ; \; Z_i \in Gen(G), \; i=1..k$$

Vì $X \subseteq Y$ nên $X \subseteq Z_i$, $i=1..k$. Theo định nghĩa của phần tử cực đại xét trong $Gen(G)$ ta suy ra $X = Z_i$, $i=1..k$, và do đó $X = Y$. Điều này chứng tỏ X là phần tử cực đại trong $G\setminus\{U\}$, $X \in MAX(G\setminus\{U\})$.

Đảo lại, giả sử $X \in MAX(G\setminus\{U\})$. Khi đó, theo định nghĩa của tập sinh, X được biểu diễn qua một giao của các phần tử trong $Gen(G)$,

$$X = V_1 \cap V_2 \cap ... \cap V_h; \; V_i \in Gen(G), \; i=1..h$$

Hệ thức trên cho ta $X \subseteq V_i$, $i=1..h$. Theo tính chất của phần tử cực đại xét trong $G\setminus\{U\}$ ta suy ra $X = V_i$, $i=1..h$, nghĩa là $X \in Gen(G)$. Theo mệnh đề trên, $X \in MAX(Gen(G))$ ■

Định nghĩa 3.2.

Cho G là một giàn giao trên tập hữu hạn U. Ta ký hiệu Coatom(G) = MAX(G \ {U}) và gọi các phần tử trong Coatom(G) là đối nguyên tử của giàn giao G.

Định lý 3.1. (Đặc trưng của các Coatom trong giàn giao)

Với mọi giàn giao G trên tập hữu hạn U ta có Coatom(G) = MAX(Gen(G)).

Mệnh đề 3.3.

Tập mục thường xuyên P của một bảng giao tác T là một giàn giao trên tập hữu hạn các mục (item) U.

Gen(P) = {X ∈ P| d(X) ≤ 1} - U, trong đó $d(X) = ||\{Y \in G \mid X \subset Y\}||$

Chứng minh:

Theo mệnh đề 3.1, tập mục thường xuyên P là một họ con trên tập hữu hạn các mục U nên P là một giàn. Mặt khác, ta cũng có P đóng với phép giao. Thật vậy, giả sử $X, Y \in P$, $Z = X \cap Y$. Ta có $Z \subseteq X$, do đó $\alpha(Z) \geq \alpha(X) \geq \delta$, suy ra $Z \in P$. Vậy, P là một giàn giao trên U ■

Mệnh đề 3.3 cho phép chúng ta vận dụng các tính chất của giàn giao trong xử lý các luật kết hợp. Cụ thể là khi cần ẩn luật kết hợp có chứa tập mục H ta sẽ sửa các tập mục lớn nhất chứa H trong giàn giao P, tức là các coatom chứa H.

Mệnh đề 3.4.

Với mỗi tập con X trong U, Poset(X) làm thành một giàn giao đầy đủ với tập Gen gồm các phần tử trên hàng thứ 2.

Chứng minh:

Giả sử $X \in P$ và $Y \subseteq X$. Ta có ngay $\alpha(Y) \geq \alpha(X) \geq \delta$. Từ đây suy ra $Y \in P$, nghĩa là mọi tập con của X đều là tập mục thường xuyên. Do *Poset*(X) chứa mọi tập con của X nên *Poset*(X) là đầy đủ và đương nhiên đóng với phép giao. Theo mệnh đề 2.3.1.1 ta thấy với mọi mục $A \in X$, $X - \{A\}$ chỉ khuyết duy nhất một phần tử, do đó chúng có duy nhất một cha. Mọi tập con còn lại trong *Poset*(X) đều khuyết từ hai phần tử trở lên do đó chúng có ít nhất là hai cha. Vậy *Gen*(X) bao gồm các phần tử đứng trên hàng thứ hai trên đồ thị biểu diễn giàn đã cho ■

Mệnh đề 3.4.

$\forall X, Y \subseteq U, X \subseteq Y$ ta luôn có $\alpha(X) \geq \alpha(Y)$.

Chứng minh:

Theo định nghĩa, $\alpha(X)$ là số giao tác có chứa X trong bảng T. Gọi T_X là tập tất cả các giao tác chứa X, T_Y là tập tất cả các giao tác chứa Y. Theo định nghĩa giao tác với tập mục bất kỳ $Z \subseteq X$ nếu $Z \in T_Y \Rightarrow Z \in T_X \Rightarrow T_X \supseteq T_Y$ hay $\alpha(X) \geq \alpha(Y)$ ▪

Mệnh đề 3.5.

Nếu H là tập thường xuyên thì mọi tập con của H (trong Poset(H)) cũng là các tập mục thường xuyên.

Chứng minh:

Giả sử X là tập con bất kỳ của tập mục thường xuyên H, $X \subseteq H$. Theo mệnh đề 2.3.1.3 ta có $\alpha(X) \geq \alpha(H)$. Vì H là tập thường xuyên nên $\alpha(H) \geq \delta$, trong đó δ là ngưỡng hỗ trợ tối thiểu. Do đó, $\alpha(X) \geq \delta$, nên theo định nghĩa thì X cũng là tập thường xuyên ▪

Mệnh đề 3.6.

Nếu tập mục phổ biến H bị ẩn thì mọi tập mục (thường xuyên) chứa H cũng phải bị ẩn theo

Chứng minh:

Để ẩn tập mục Y nào đó, ta phải làm cho $\alpha(Y) < \delta$. Gọi Y là một tập mục bất kỳ có chứa tập mục thường xuyên có chứa H. Khi đó $\alpha(H) \geq \alpha(Y)$. Do đó, nếu ẩn H thì $\delta > \alpha(H) \geq \alpha(Y)$. Vì vậy Y cũng bị ẩn đi ▪

Để ẩn tập mục H mà số giao tác trong bảng dữ liệu không thay đổi, một trong những phương pháp là sửa giá trị của một trong các mục $A \subseteq H$ từ 1 thành 0.

Gọi hàm Update(A,X) là sửa mục A từ 1 thành 0 trên một giao tác chứa X và Update(A,X,n) là sửa mục A từ 1 thành 0 trên n giao tác chứa X trong bảng dữ liệu giao tác. Ta có mệnh đề sau:

Mệnh đề 3.7.

Update(A,X) $\Rightarrow$ --$\alpha(Y)$, $Y \in Poset(X)$, $A \in Y$

Chứng minh:

Ta có Update(A,X) = Update(t,A), X = Set(t) => $A \subset X$. Do Update(t,A) là sửa A trên giao tác t từ 1 thành 0, nên sau khi thực hiện hàm Update(A,X) thì $A \not\subset$Set(t). Gọi Y là tập mục bất kỳ trong Poset(X) và $A \subset Y$, vì $Y \subset X$, nên Update(A,X) kéo theo A$\not\subset$Y trên giao tác t, do đó $\alpha(Y)$ giảm đi 1. Vậy, Update(X,A) => -- $\alpha(Y)$ ▪

Các kết quả trên cho thấy vận dụng lý thuyết giàn giao có thể tìm ra cận trên đúng và cận dưới đúng của tập mục cần giảm độ hỗ trợ nhằm mục đích ẩn luật. Từ đó, xác định các tập mục chịu ảnh hưởng trực tiếp khi ẩn luật, làm cơ sở để đề xuất các heuristic nhằm tránh tối đa sự tác động lên các tập mục này dẫn đến việc giảm tối đa các hiệu ứng phụ gây ra trong quá trình ẩn.

4. Phương pháp ẩn luật kết hợp dựa trên tiếp cận giàn giao

Trong 2 hướng tiếp cận để ẩn luật kết hợp đã đưa ra trong mục 2) thì tiếp cận thứ 2 được xem là tốt hơn trong tình huống muốn giảm tối đa các luật bị ẩn nhầm. Xét luật R=>S cần được ẩn. Độ tin cậy của luật là $\beta(R$=>$S) = \alpha(RS)/\alpha(R)$. Giảm độ tin cậy của luật xuống dưới ngưỡng tin cậy tối thiểu σ bằng cách sửa mục $A \subset S$ trên các giao tác chứa RS. Trong giàn giao các tập mục thường xuyên P (mệnh đề 3.3), nếu rút một nút khỏi giàn (ẩn tập mục) thì sẽ dẫn đến nguy cơ các tập mục khác ở mức dưới (có độ hỗ trợ thấp hơn) bị ảnh hưởng, do đó có thể gây ra hiệu ứng phụ. Vấn đề đặt ra là sửa A trên các giao tác nào và sửa bao nhiêu lần là đủ để ẩn luật mà các hiệu ứng phụ gây ra là ít nhất.

Mệnh đề 3.4 và tính chất nghịch biến của hàm α cho ta thấy rằng các phần tử trong $Gen(X)$ có độ hỗ trợ nhỏ nhất trong $Poset(X)-\{X\}$. Nếu $X \in Coatom(P)$, với mỗi mục $A \in S$ trong X ta xét hàm $L(A,X)$ cho giá trị là bộ ba (A,Y,λ) trong đó λ là giá trị nhỏ nhất trong số các độ hỗ trợ của các tập con đúng Y chứa A của X (tức là $Y \subseteq X$, $Y \neq X$ và $A \subset Y$),

$L(A,X) = \{A,Y,\lambda\}$, $\lambda = min\ \{\alpha(Y) \mid A \subset Y, Y \in \text{Gen}(X), X \in Coatom(P)\}$

Dựa vào nhận xét trên ta thấy có thể tính $L(A,X)$ thông qua các tập chứa A trong $Gen(X)$.

Ta thấy rằng các tập mục trong $L(A,X)$ có nguy cơ bị giảm độ hỗ trợ và dẫn đến ẩn nhầm luật là cao nhất khi thay đổi giá trị của A. Rõ ràng những tập mục trong $L(A,X)$ là những tập mục thường xuyên cần được bảo vệ trong suốt quá trình ẩn. Để tốc độ hội tụ của thuật toán nhanh và duy trì được tập $L(A,X)$ thì việc sửa đổi cần phải được thực hiện trên các giao tác chứa các tập mục có độ hỗ trợ lớn nhất trong số các tập mục Y thu được từ các L(A,X), tức là lựa chọn tập mục đạt $max\{\alpha(Y) \mid Y \in L(A,X)\}$.

Gọi tập các tập mục chứa S trong $Coatom(P)$ là V , khi đó

$V = \{X \in Coatom(P) \mid RS \subseteq X\}$

Với mỗi mục $A \subset S$ và với mỗi tập mục X trong V lượng giá xem có nên sửa mục A trong X không? Tiêu chuẩn đặt ra là việc sửa mục A trong X không gây hiệu ứng phụ đến các tập con đúng chứa A của X. Gọi $M(S)$ là hàm cho giá trị là bộ tứ (A, X, Z, μ) trong đó μ là giá trị lớn nhất trong số các độ hỗ trợ λ tìm được qua các hàm $L(A,X)$, cụ thể là,

$M(S) = (A, X, Z, \mu)$, $\mu = max\ \{\lambda \mid L(A,X) = (A,Z,\lambda), A \subset S, X \in V\}$.

Hàm $M(S)$ cho ta biết rằng cần phải sửa mục A trên các giao tác chứa $X \in V$, mục A đạt giá trị *maxmin* theo độ hỗ trợ tại tập mục Z/μ.

Trong bài toán ẩn luật kết hợp, giảm số lần truy cập dữ liệu là một tiêu chí rất được quan tâm nhằm giảm độ phức tạp về thời gian của thuật toán. Bước tiếp theo của thuật toán là tính xem với mỗi mục ứng viên ẩn $A \in S$ tìm được cần sửa bao nhiêu lần trên các giao tác chứa tập mục $X \in V$. Trước hết, ta xác định xem sửa A bao nhiêu lần để có thể ẩn luật. Ta có $\beta(R=>S) = \alpha(RS)/\alpha(R)$, gọi $q = \lceil \sigma * \alpha(R) \rceil$, để $\beta(R=>S) < \sigma$ hay $\alpha(RS)/\alpha(R) < \sigma$ suy ra số lần ít nhất cần sửa A để ẩn luật là $\alpha(RS) - q$. Tuy nhiên, vì cần phải bảo vệ các tập mục thường xuyên không thuộc luật nhạy cảm nhằm tránh ẩn nhầm luật phổ biến không nhạy cảm, mà mục ứng viên ẩn A đạt *maxmin* tại Z/μ; đồng thời cần khai thác tối đa tập mục chứa S trong coatom(P) để tiết kiệm số lần truy cập bảng giao tác, nên số lần sửa mục ứng viên trong mỗi lần truy cập bảng giao tác là: $n:=\max\{\min\{\alpha(RS) - q, \alpha(Z) - \delta, \alpha(X)\}, 1\}$. Sở dĩ lấy $n=\max\{\min\{\ldots\}, 1\}$ là để tránh vấp phải vòng lặp vô hạn khi hàm min cho giá trị bằng 0.

Quá trình trên được lặp lại cho đến khi $\beta(R=>S) < \sigma$.

Thuật toán HidingRule được mô tả bằng ngôn ngữ thuật toán:

Algorithm HidingRule(P,δ,R=>S)

Input: *T – bảng trị 0/1 các giao tác trên tập mục nền U; P – họ các tập mục thường xuyên của U; σ – ngưỡng tin cậy tối thiểu; $R=>S$ – luật nhạy cảm (cần ẩn).*

Output: *bảng kết quả T có thể khai thác được các luật kết hợp phổ biến, trừ luật $R=>S$.*

Method

$d := \alpha(RS)/\alpha(R)$;

Compute $V = \{X \in Coatom(P) \mid RS \subseteq X\}$;

k := $||S||$;

while (d > σ – 1) **do**

// Chọn ứng viên ẩn.

Let $(A,X,Z,\mu) = M(S)$;

k := k-1; q := q = $\lceil \sigma^* \alpha(R) \rceil$;

For $X \in V$ **Do**

If $\alpha(X)=0$ **Then** Continue;

n:=max{min{$\alpha(RS) - q, \alpha(Z) - \delta, \alpha(X)$},1};

Update(*A,X,n*);

d := $\alpha(RS)/\alpha(S)$;

If d<σ **Then** Break;

EndFor;

If d<σ **Then** Break;

If k=0 Then *// đã chọn hết ứng viên ẩn*

Compute $V = \{X \in Coatom(P) \mid RS \subseteq X\}$;

k := ||*S*||;

EndIf;

Endwhile;

HidingRule.

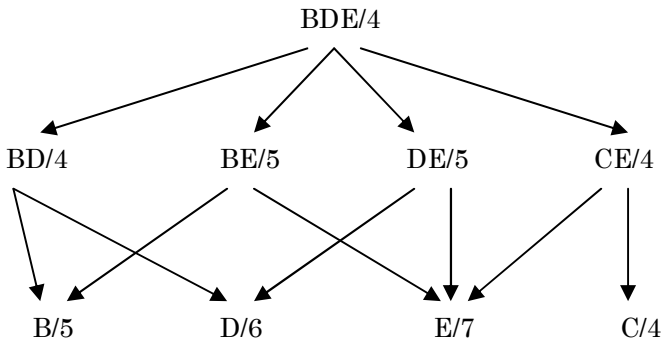
Ví dụ 2: Xét bảng giao tác cho trong ví dụ 1. Khi đó, tập luật phổ biến có được với ngưỡng hỗ trợ tối thiểu $\delta = 4$ và ngưỡng tin cậy tối thiểu $\sigma = 70\%$ là: $R = \{B \Rightarrow D, B \Rightarrow E, D \Rightarrow E, E \Rightarrow D, E \Rightarrow B, B \Rightarrow DE, BD \Rightarrow E, BE \Rightarrow D, DE \Rightarrow B\}$. Giả sử luật cần ẩn là $B \Rightarrow E$. Thuật toán được thực hiện như sau:

Khởi tạo giá trị: d = 100%. V:={*BDE*}. k:=1; q:=[70%*5] = 3

Lặp: d>σ-1. Tính L(*E*,*BDE*) = {*E*, *BE*, 5}; L(*E*,*CE*) = {*E*, *CE*, 4};

=> Let(*A*,*X*,*Z*,μ)= (*E*, *BDE*, *BE*, 5). Vậy, cần sửa *E* trong các giao tác chứa tập *V*, đạt max tại *BE*/5.

- Với *X* = *BDE*, ta tính: n = max{min{5-3, 1, 4},1} = 1. Sửa *E* trên 1 giao tác đầu



Hình 3.1. *Giàn giao P gồm các tập mục / độ hỗ trợ*

tiên có chứa *BDE*, giao tác có số hiệu 1. d = 4/5*100% = 80% >σ.

- k=0. Tính lại tập *V* = {*BE*}. Với *X* = *BE*, tính n=max{min{4-3,0,4},1} = 1. Sửa *E* trên giao tác đầu tiên có chứa *BE*, giao tác có số hiệu 2. Lúc này d = 3/5 * 100% = 60% < σ. Vậy luật *B*=>*E* được ẩn.

Tập luật sau khi ẩn là: *R* = {*B*=>*D*, *D*=>*E*, *E*=>*D*}.

Định lí 4.1. *Thuật toán HidingRule là đúng đắn.*

Định lí 4.2. *Thuật toán HidingRule có độ phức tạp là đa thức.*

5. Kết luận

Bài báo này đã đề xuất một tiếp cận mới để giải quyết bài toán ẩn luật kết hợp nhạy cảm. Thuật toán HidingRule được đề xuất dựa trên tiếp cận lý thuyết giàn giao, có độ phức tạp là đa thức. Với số mục vừa phải, chẳng hạn 64 mục, thì thuật toán có thể được cài đặt được với việc quản lý mục thông qua các số nguyên. Với độ phức tạp đa thức, thuật toán có thể đưa ra cài đặt ứng dụng với những bảng cơ sở dữ liệu có số lượng bản ghi là vài chục ngàn. Nếu lựa chọn cấu trúc dữ liệu thích hợp, thuật toán có thể áp dụng được cho các bảng dữ liệu lớn.

TÀI LIỆU THAM KHẢO

1. Nguyễn Xuân Huy, *Các phụ thuộc logic trong cơ sở dữ liệu*, NXB Thống kê, 2006.
2. Alena Dasseni, Vassilios S. Verykios, Ahmed K. Elmagarmid, Elisa Bertino, *Hiding Association Rules by Using Confidence and Support*, Proceeding of the 14th International Workshop on Informationn Hiding, (2001), 369-383.
3. George V. Moustakides, Vassilios S. Verykios, *A MaxMin Approach for Hiding Frequent Itemsets*, Data & Knowledge Engineering, 65, (2008), 75-89.
4. Verykos, V. Elmagarmid, A. Bertino, E. Saygin, Dasseni, *Association Rules Hiding, IEEE Transaction on Knowledge and Data Engineering,* (2004), 434-447.
5. Xingzhi Sun, Philip S. Yu, *Hiding Sensitive Frequent Itemsets by a Border-Based Approach*, Computing and Engineering, v. 1 n.1, (2007), 74-94.

ASSOCIATION RULE HIDING METHOD BASED ON INTERSECTION LATTIC APPROACH

Le Quoc Hai
Quang Tri College of Education

SUMMARY

Association rule hiding is an important problem in terms of the data mining.One of the issues raised when solving this problem is to reduce side effects, that are hidden off the rules leading to new rules being born, and to reduce the number of data access. The report introduces a new approach based on theoretical intersection lattic. Hiding Rules algorithm obtained is strictly mathematical basis, using the heuristic to identify the items, the transactions need to hide the sensitive association rules so as to result in fewest side effects.