

**BỘ CÔNG THƯƠNG**

**TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP THỰC PHẨM TP.HCM**

**KHOA CÔNG NGHỆ THÔNG TIN**

🙜🙢🙠🙞



**KHÓA LUẬN TỐT NGHIỆP**

**ĐỀ TÀI: NGUYÊN CỨU THUẬT TOÁN HUI-MINER VÀ XÂY DỰNG ỨNG DỤNG TÌM TẬP MỤC HỮU ÍCH CAO TRONG SIÊU THỊ BIGC**

Giảng viên hướng dẫn : Cô Nguyễn Thị Thanh Thủy

Sinh viên thực hiện :

1. Phạm Văn Hiệp - 2001160224

2. Phan Văn Thắng - 2001160340

TP. HỒ CHÍ MINH .. – 2020

**LỜI CẢM ƠN**

Để bài báo cáo của nhóm đạt được kết quả tốt, nhóm em đã nhận được sự giúp đỡ, hỗ trợ từ nhiều thầy cô, bạn bè và gia đình. Với sự chân thành, tình cảm sâu sắc, cho phép nhóm em được bày tỏ lời cảm ơn sâu sắc đến những người đã hỗ trợ nhóm em trong thời gian làm đồ án vừa qua.

Nhóm em xin gửi lời cảm ơn chân thành đến cô Nguyễn Thị Thanh Thủy, giảng viên ngành công nghệ thông tin – trường Đại học công nghiệp thực phẩm Hồ Chí Minh cô đã hướng dẫn tận tình và chỉ bảo cho nhóm em trong quá trình làm khóa luận.

Với điều kiện thời gian cũng như kinh nghiệm còn hạn chế của nhóm nên bài báo cáo này sẽ không trách được những thiếu sót. Nhóm em rất mong nhận được chỉ bảo, đóng góp ý kiến của các thầy cô để nhóm em có điều kiện bổ sung, nâng cao kiến thức.

Nhóm em xin chân thành cảm ơn !

MỤC LỤC

[**CHƯƠNG 1: TÌM HIỂU VỀ TẬP PHỔ BIẾN VÀ LUẬT KẾT HỢP** 3](#_Toc38006031)

[**1.1.** **TẬP PHỔ BIẾN** 3](#_Toc38006032)

[**1.1.1.** **ĐỊNH NGHĨA** 3](#_Toc38006033)

[**1.1.2.** **ỨNG DỤNG** 4](#_Toc38006034)

[**1.1.3.** **TÍNH CHẤT** 4](#_Toc38006035)

[**1.2.** **LUẬT KẾT HỢP** 4](#_Toc38006036)

[**1.2.1.** **GIỚI THIỆU** 4](#_Toc38006037)

[**1.2.2.** **NỘI DUNG** 4](#_Toc38006038)

[**1.2.3.** **ĐỊNH NGHĨA VỀ LUẬT KẾT HỢP** 5](#_Toc38006039)

[**1.2.4.** **TÍNH CHẤT** 6](#_Toc38006040)

[**1.2.5.** **KHAI PHÁ LUẬT KẾT HỢP** 6](#_Toc38006041)

[**CHƯƠNG 2: TÌM HIỂU VỀ KHAI THÁC TẬP HỮU ÍCH CAO** 7](#_Toc38006042)

[**2.1.** **CÁC KHAI NIỆM, MỤC ĐICH, ĐỊNH NGHĨA LIÊN QUAN** 7](#_Toc38006044)

[**2.1.1.** **KHÁI NIỆM** 7](#_Toc38006045)

[**2.1.2.** **MỤC ĐÍCH** 7](#_Toc38006046)

[**2.1.3.** **ĐỊNH NGHĨA** 7](#_Toc38006047)

[**2.2.** **TÌM HIỂU THUẬT TOÁN HUI-MINER ĐỂ KHAI THÁC TẬP HỮU ÍCH CAO** 9](#_Toc38006048)

[**2.2.1.** **GIỚI THIỆU THUẬT TOÁN** 9](#_Toc38006049)

[**2.2.2.** **THUẬT TOÁN HUI- MINER** 9](#_Toc38006050)

[**2.2.3.** **KẾT LUẬN** 18](#_Toc38006051)

[**TÀI LIỆU THAM KHẢO** 19](#_Toc38006052)

**DANH MỤC CÁC KÝ HIỆU VÀ CHỮ VIẾT TẮT**

**Viết tắt Tiếng Anh Tiếng Việt**

**CSDL Data Base(DB) Cơ sở dữ liệu**

**HUI High utility itemset Tập mục lợi ích cao**

**DANH MỤC CÁC HÌNH VẼ VÀ ĐỒ THỊ**

**DANH MỤC CÁC BẢNG BIỂU**

**NHẬN XÉT CỦA GIÁO VIÊN**

**..........................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................**

**..........................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................**

**..........................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................**

**..................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................**

**MỞ ĐẦU**

Hiện nay, việc tính toán doanh số và tối ưu hóa lợi nhuận bán hàng là công việc cực kỳ quan trọng, nó ảnh hưởng trực tiếp đến doanh thu và chiến lược bán hàng của các công ty, siêu thị hay các đơn vị bán lẻ. Đặc biệt, với số lượng hàng hóa lớn, giá cả khác nhau, nên việc tính toán lợi nhuận tối ưu từ bán hàng càng có quan trọng. Trong khi số lượng giao dịch mỗi giờ có thể lên đến hàng chục nghìn giao dịch, việc tính toán xem mặt hàng nào đem lại doanh số cao, mặt hàng nào kinh doanh không hiệu quả dù bán với số lượng lớn càng trở nên khó khăn do dữ liệu quá lớn, liên tục.

Bài toán khai thác tập mục hữu ích cao(High-Utility Itemsets Mining – HUIM) đã được nhóm tác giả R.C. Chan, Q. Yang, Y.D. Shen đề xuất vào năm 2003, để tìm ra các HUI(High-Utility Itemsets), là các tổ hợp đem lại lợi nhuận cao nhất từ cơ sở dữ liệu giao dịch được lưu lại. Từ đó, các công ty, siêu thị bán lẻ sẽ đưa ra các chiến lược kinh doanh cho phù hợp, nhằm tối đa hóa lợi nhuận.

Chính vì những lợi ích to lớn mà tập mục hữu ích cao mang lại nên nhóm chúng em đã lựa chọn đề tài “Nghiên cứu thuật toán HUI-Miner và xây dựng ứng dụng tìm tập mục hữu ích cao trong siêu thị BigC”. Với mục tiêu chính là đưa ra được danh sách những sản phẩm mang lại lợi nhuận cao, đồng thời từ danh sách sản phẩm mang lại lợi nhuận cao đó sẽ dự báo các sản phẩm cần nhập để đem lại lợi ích tốt nhất cho siêu thị.

Đối tượng và phạm vi đề tài: tìm hiểu, nguyên cứu và thu thập thông tin từ chuỗi cửa hàng siêu thị BigC trên địa bàn thành phố Hồ Chí Minh.

**CHƯƠNG 1: TÌM HIỂU VỀ TẬP PHỔ BIẾN VÀ LUẬT KẾT HỢP**

* 1. **TẬP PHỔ BIẾN**
     1. **ĐỊNH NGHĨA**
* **Độ phổ biến:**

Cho một cơ sở dữ liệu giao dịch K, với A là một tập con của T (A T), độ phổ biến của A trong K tức là số lần A xuất hiện trong các giao dịch của cơ sở dữ liệu giao dịch K, ký hiệu .

* **Tập phổ biến**

hay còn gọi là mẫu phổ biến (tập các hạng mục, chuỗi con, cấu trúc con, đồ thị con,..) được gọi là phổ biến nếu độ phổ biến của nó lớn hơn hoặc bằng MinSup (với MinSup là một giá trị cho trước)

Ví dụ minh họa:

|  |  |
| --- | --- |
| Mã giao dịch | Nội dung giao dịch |
| 1 | A,B,C,F |
| 2 | A,C,D,E,F |
| 3 | A,C,E,F |
| 4 | A,B,D,E |
| 5 | B,C,D |

Tính độ phổ biến:

, , , , ,

Với MinSup = 3 (tức là 60%) thì sẽ có các mục thỏa mãn là: A, C, E

Sau đó ta sẽ tìm các mục kết hợp có độ phổ biến thỏa yêu cầu như sau:

|  |  |
| --- | --- |
| Danh mục | Độ phổ biến |
| AC | 3 |
| AE | 3 |
| CE | 2 |

Vì nên loại CE

Tiếp tục với mức 3 danh mục:

|  |  |
| --- | --- |
| Danh mục | Độ phổ biến |
| ACE | 2 |

* Dừng lại.
* Kết luận: các mục thoản mãn yêu cầu đặt ra là: A, C, E, AC, AE
  + 1. **ỨNG DỤNG**
* Áp dụng trong phân tích CSDL bán hàng
* Mở rộng sang quảng cáo, thiết kế catalog, phân tích chiến dịch bán hàng, Web log, chuỗi AND,..
  + 1. **TÍNH CHẤT**

**Tính chất 1:** mọi tập con của tập phổ biến đều phổ biến, nếu , =>

**Tính chất 2:** mọi tập cha của tập không phổ biến đều không phổ biến, nếu , => . Tính chất này quan trọng để sau này ứng dụng trong thuật toán.

* 1. **LUẬT KẾT HỢP**
     1. **GIỚI THIỆU**

Trong lĩnh vực Data Mining, mục đích của luật kết hợp (Association Rule - AR) là tìm ra các mối quan hệ giữa các đối tượng trong khối lượng lớn dữ liệu.

* + 1. **NỘI DUNG**

Cho cơ sở dữ liệu gồm các giao dịch T là tập hợp các giao dịch t1, t2, … , tn

Mỗi giao dịch gồm các đối tượng I (gọi là Itemset).

I = {i1, i2, … , im} mỗi itemset gồm k item gọi là k-itemset

Mục đích của luật kết hợp là tìm ra sự kết hợp (association) hay tương quan (correlation) giữa các items. Những luật kết hợp này có dạng***X -> Y*, luật kết hợp này có thể được hiểu đơn giản là khi người ta mua các sản phẩm trong tập X thường mua các sản phẩm trong tập Y.**

**Ví dụ: X = {Sữa, thịt}, Y = {Bánh mì, rau} và ta có luật kết hợp X -> Y thì chúng ta sẽ hiểu là những người mua sữa và thịt thì sẽ thường mua kèm bánh mì và rau.**

**Trong thống kê, X được xem là biến độc lập còn Y là biến phụ thuộc.**

* + 1. **ĐỊNH NGHĨA VỀ LUẬT KẾT HỢP**
* **Định nghĩa**

Cho I={I1, I2, .., Im} là tập hợp của m tính chất riêng biệt. Giả sử D là CSDL, với các bản ghi chứa một tập con T các tính chất (có thể coi như Τ ⊆ Ι), các bản ghi đều có chỉ số riêng. Một luật kết hợp là một mệnh đề kéo theo có dạng X→Y, trong đó X, Y ⊆ I, thỏa mãn điều kiện X∩Y=∅. Các tập hợp X và Y được gọi là các tập hợp tính chất (itemset). Tập X gọi là nguyên nhân, tập Y gọi là hệ quả. Có 2 độ đo quan trọng đối với luật kết hợp: Độ hỗ trợ (support) và độ tin cậy (confidence), được định nghĩa như phần dưới đây.

* **Định nghĩa độ hỗ trợ**

Độ hỗ trợ (Support) của luật kết hợp X ->Y là tần suất của giao dịch chứa tất cả các items trong cả hai tập X và Y. Ví dụ, support của luật X ->Y là 5% có nghĩa là  5% các giao dịch X và Y được mua cùng nhau

Công thức để tính support của luật X =>Y như sau:

Support (X -> Y) = P(X ) =

Trong đó: N là tổng số giao dịch.

n(X ) là số giao dịch chứa cả X và Y.

* **Định nghĩa độ tin cậy**

Độ tin cậy (Confidence) của luật kết hợp X ->Y là xác suất xảy ra Y khi đã biết X. Ví dụ độ tin cậy của luật kết hợp {Sữa} ->Bánh mì} là 80% có nghĩa là 80% khách hàng mua Apple cũng mua Banana.

Confidence (X -> Y) = P(Y ) =

Trong đó: n(X) là số giao dịch chứa X

Độ tin cậy của một luật kết hợp X Y là tỷ lệ giữa số lượng các bản ghi của tập hợp chứa X Y, so với tổng số các bản ghi chứa X.

* **Định nghĩa tập thường xuyên**

Tập hợp X được gọi là tập hợp thường xuyên (Frenquent itemset) nếu có supp(X) minsup, với minsup là ngưỡng độ hỗ trợ cho trước. Kí hiệu các tập này là FI.

* + 1. **TÍNH CHẤT**

**Tính chất 1:** Giả sử A,B I là hai tập hợp với A B thì supp(A) supp(B).

Như vậy, những bản ghi nào chứa tập hợp B thì cũng chứa tập hợp A

**Tính chất 2:** Giả sử A, B là hai tập hợp, A,B I, nếu B là tập hợp thường xuyên và A B thì A cũng là tập hợp thường xuyên. Nếu tập B là tập thường xuyền tức là supp(B) MinSup và A B: Supp(A) supp(B) (tính chất 1) A cũng là tập thường xuyên (supp(A) supp(b) MinSup)

**Tính chất 3:** Giả sử A, B là hai tập hợp, A B và A là tập hợp không thường xuyên thì B cũng là tập hợp không thường xuyên.

**Để thu được các luật kết hợp, ta thường áp dụng 2 tiêu chí:** minimum support (min\_sup) và  minimum confidence (min\_conf)

Các luật thỏa mãn có support và confidence thỏa mãn (lớn hơn hoặc bằng)  cả Minimum support và Minimum confidence gọi là các luật mạnh (Strong Rle)

Minimum support và Minimum confidence gọi là các giá trị ngưỡng (threshold) và phải xác định trước khi sinh các luật kết hợp.

Một itemsets mà tần suất xuất hiện của nó >= min\_sup gọi là frequent itemsets

* + 1. **KHAI PHÁ LUẬT KẾT HỢP**

Khai phá luật kết hợp là công việc phát hiện ra (tìm ra, khám phá, phát hiện) các luật kết hợp thỏa mãn các ngưỡng độ hỗ trợ và ngưỡng độ tin cậy cho trước. Bài toán khai phá luật kết hợp được chia thành hai bài toán nhỏ, hay như người ta thường nói, việc giải bài toán trải qua hai bước:

Bước 1: Tìm tất cả các tập phổ biến (tìm FI) trong CSDL D.

Bước 2: Tạo ra các luật kết hợp mạnh từ các tập phổ biến.

Hiệu suất của quá trình khai thác dữ liệu phụ thuộc rất lớn vào bước đầu tiên.

1. Thuật toán Apriori
2. Thuật toán FP-Growth

**CHƯƠNG 2: TÌM HIỂU VỀ KHAI THÁC TẬP HỮU ÍCH CAO**

1. 1. **CÁC KHAI NIỆM, MỤC ĐICH, ĐỊNH NGHĨA LIÊN QUAN**
      1. **KHÁI NIỆM**

Cơ sở dữ liệu dạng giao tác (Trasactional database): Là dạng cơ sở dữ liệu tác nghiệp nhưng các bản ghi thường là các giao tác. Dạng dữ liệu này thường phổ biến trong lĩnh vực thương mại và ngân hàng.

* + 1. **MỤC ĐÍCH**

Khai thác tập mục hữu ích cao là làm giảm thiểu kích thước của tập ứng viên và làm đơn giản hóa quá trình tính toán độ hữu ích các tập mục từ đó giảm số lượng ứng viên cho tập mục hữu ích cao, giảm thời gian khai thác.

* + 1. **ĐỊNH NGHĨA**

**Định nghĩa 1:** Lợi ích giao tác (TU) của một giao tác Tc là tổng lợi ích của các các danh mục từ Tc trong Tc, tức là TU(Tc) =

**Ví dụ:** TU(T1) = 1\*5+1\*1+1\*2 = 8

**Định nghĩa 2:** Lợi ích trọng số giao dịch (TWU) của 1 itemset X được định nghĩa bằng tổng lợi ích giao tác của các giao tác chứa X, tức là:

TWU(X) =

**Ví dụ:** TWU(A) = TU(T1) +TU(T2)+TU(T3) = 8+27+30 = 65

**Định nghĩa 3:** Giá trị tiện ích tối thiểu được chỉ định là giá trị tiện ích tối thiểu tuyệt đối, được ký hiệu là minutil và được tính như sau:

Minutil =

**Ví dụ:** Giả sử , khi đó minutil được tính như sau:

Minutil = \* (TU(T1) + TU(T2) + TU(T3) + TU(T4) + TU(T5)

= 28% \* (8 + 27 + 30 + 20 + 11) = 0.28 \* 96 = 26,88

**Định nghĩa 4:** Tiện ích của một danh mục X còn lại trong giao dịch Tc(X ) được ký hiệu là RU(X,Tc) hay còn được viết là Rutil được tính là:

RU(X,Tc) =

**Ví dụ:** RU(C,T1) = u(A)+u(D) = 5+2 = 7

**Định nghĩa 5:** Tiện ích của mục X, ký hiệu u(X) là tổng tiện ích của X trong tất cả các giao dịch có chứa X trong D trong đó u(X) sẽ được tính như sau:

u(X) =

**Ví dụ:** u(A) = u(A,T1) + u(A,T2) + u(A,T3) = 5+10+5 = 20

**Định nghĩa 6:** Một giao dịch được coi là đã được sửa đổi là sau khi tất cả các mục có tiện ích có trọng số giao dịch nhỏ hơn một minutil đã cho sẽ bị xóa khỏi giao dịch; các mục còn lại được sắp xếp theo thứ tự tăng dần theo trọng số giao dịch.

**Ví dụ:** Giả sử cho minutil = 60 và giá trị lợi ích trọng số giao dịch (TWU) lần lượt là:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Danh mục | A | B | C | D | E | F | G |
| TWU | 65 | 61 | 96 | 58 | 88 | 30 | 38 |

Với giá trị minutil >= 40 từ các danh mục bị loại ra khỏi danh sách là D, F, G. Ta tiến hành sắp xếp lại các danh mục theo thứ tự từ nhỏ đến lớn theo giá trị TWU và sẽ được là: **B < A < E < C**

Khi đó bảng lợi ích sẽ được sửa đổi như sau:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Tid** | Item Util | Item Util | Item Util | Item Util | TU |
| **T1** | A 5 | C 1 |  |  | 6 |
| **T2** | A 10 | C 6 |  |  | 16 |
| **T3** | B 4 | A 5 | E 3 | C 1 | 13 |
| **T4** | B 8 | E 3 | C 3 |  | 14 |
| **T5** | B 4 | E 3 | C 2 |  | 9 |

**Định nghĩa 7:** Cho một mục X và một giao dịch (hoặc mục) T với X⊆T, tập hợp tất cả các mục sau X trong T được ký hiệu là T / X.

**Ví dụ:** T1/{C} = {AD}

* 1. **TÌM HIỂU THUẬT TOÁN HUI-MINER ĐỂ KHAI THÁC TẬP HỮU ÍCH CAO**
     1. **GIỚI THIỆU THUẬT TOÁN**

Để xác định tập lợi ích cao, thuật toán đầu tiên tạo ra tập ứng cử viên từ cách đánh giá các lợi ích cao và sau đó tính toán các lợi ích chính xác của các ứng cử viên. Các thuật toán khác gặp những vấn đề là tạo ra một số lượng lớn tập ứng viên nhưng hầu hết các ứng cử viên được sinh ra là lợi ích không cao sau khi các lợi ích được tính chính xác. HUI- Miner (High Utility Itemset Miner) sử dụng một cấu trúc mới, được gọi là danh sách lợi ích, để lưu trữ tất cả các thông tin hữu ích về một tập và tìm ra thông tin để cắt tỉa không gian tìm kiếm của HUI- Miner. Bằng cách tránh tạo ra các tập ứng viên thế hệ và tính toán lợi ích của nhiều tập ứng viên, HUI- Miner hiệu quả hơn vì có thể khai thác tập lợi ích cao từ danh sách lợi ích (utility list)

* + 1. **THUẬT TOÁN HUI- MINER**

|  |  |
| --- | --- |
| Tid | Các giao dịch |
| T1 | (A,1) (C,1) (D,1) |
| T2 | (A,2) (C,6) (E,2) (G,5) |
| T3 | (A,1) (B,2) (C,1) (D,6) (E,1) (F,5) |
| T4 | (B,4) (C,3) (D,3) (E,1) |
| T5 | (B,2) (C,2) (E,1) (G,2) |

Bảng biển diễn giá trị lợi nhuận của các mục:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Danh mục | A | B | C | D | E | F | G |
| Lợi nhuận | 5 | 2 | 1 | 2 | 3 | 1 | 1 |

Bảng lợi ích:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Tid** | **A** | **B** | **C** | **D** | **E** | **F** | **G** |
| **T1** | 5 | 0 | 1 | 2 | 0 | 0 | 0 |
| **T2** | 10 | 0 | 6 | 0 | 6 | 0 | 5 |
| **T3** | 5 | 4 | 1 | 12 | 3 | 5 | 0 |
| **T4** | 0 | 8 | 3 | 6 | 3 | 0 | 0 |
| **T5** | 0 | 4 | 2 | 0 | 3 | 0 | 2 |

Tính giá trị lợi ích giao tác (TU) của các giao tác T1, T2, T3, T4, T5 khi thực thi là:

T1 = 5+1+2 = 8, T2 = 10+6+6+5 = 27, T3 = 5+4+1+12+3+5 = 30

T4 = 8+3+6+3 = 20, T5 = 4+2+3+2 = 11

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| TID | T1 | T2 | T3 | T4 | T5 |
| TU | 8 | 27 | 30 | 20 | 11 |

Tính lợi ích trọng số (TWU) của các danh mục khi thực thi:

A = T1+T2+T3 = 8+27+30 = 65

B = T3+T4+T5 = 30+20+11 = 61

C = T1+T2+T3+T4+T5 = 8+27+30+20+11 = 96

D = T1+T3+T4 = 8+30+20 = 58

E = T2+T3+T4+T5 = 27+30+20+11 = 88

F = T3 = 30

G = T2+T5 = 27+11 = 38

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Danh mục | A | B | C | D | E | F | G |
| TWU | 65 | 61 | 96 | 58 | 88 | 30 | 38 |

Trong thuật toán HUI-Miner mỗi mục chứa một danh sách tiện ích. Danh sách tiện ích ban đầu lưu trữ thông tin tiện ích về cơ sở dữ liệu được khai thác. Các tiện ích có trọng số giao dịch của tất cả các mục được tích lũy bằng cách quét cơ sở dữ liệu. Nếu tiện ích có trọng số giao dịch của một danh mục nhỏ hơn một minutil nhất định, thì mặt hàng đó không còn được xem xét trong quy trình khai thác tiếp theo. Đối với các mục có tiện ích có trọng số giao dịch vượt quá minutil, chúng được sắp xếp theo thứ tự tăng dần theo giao dịch có trọng số giao dịch. Đối với cơ sở dữ liệu trên, giả sử minutil là 50, và sau đó thuật toán không còn xem xét các mục **F** và **G** sau khi quét cơ sở dữ liệu đầu tiên. Các mục còn lại được sắp xếp: **D < B < A < E < C**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Tid** | Item Util | Item Util | Item Util | Item Util | Item Util | TU |
| **T1** | D 2 | A 5 | C 1 |  |  | 8 |
| **T2** | A 10 | E 6 | C 6 |  |  | 22 |
| **T3** | D 12 | B 4 | A 5 | E 3 | C 1 | 25 |
| **T4** | D 6 | B 8 | E 3 | C 3 |  | 20 |
| **T5** | B 4 | E 3 | C 2 |  |  | 9 |

Mỗi phần tử trong danh sách tiện ích của các mục X chứa ba trường: tid, iutil và rutil.

• Trường tid chỉ ra một giao dịch T có chứa X.

• Trường iutil là tiện ích của X trong T, tức là, u (X, T).

• Trường rutil là tiện ích còn lại của X trong T, tức là ru (X, T).

{D}

|  |  |  |
| --- | --- | --- |
| Tid | Iutils | Rutils |
| T1 | 2 | 6 |
| T3 | 12 | 13 |
| T4 | 6 | 14 |
| Tổng | 20 | 33 |

{B}

|  |  |  |
| --- | --- | --- |
| Tid | Iutils | Rutils |
| T3 | 4 | 9 |
| T4 | 8 | 6 |
| T5 | 4 | 5 |
| Tổng | 16 | 20 |

{A}

|  |  |  |
| --- | --- | --- |
| Tid | Iutils | Rutils |
| T1 | 5 | 1 |
| T2 | 10 | 12 |
| T3 | 5 | 4 |
| Tổng | 20 | 17 |

{E}

|  |  |  |
| --- | --- | --- |
| Tid | Iutils | Rutils |
| T2 | 6 | 6 |
| T3 | 3 | 1 |
| T4 | 3 | 3 |
| T5 | 3 | 2 |
| Tổng | 15 | 12 |

{C}

|  |  |  |
| --- | --- | --- |
| Tid | Iutils | Rutils |
| T1 | 1 | 0 |
| T2 | 6 | 0 |
| T3 | 1 | 0 |
| T4 | 3 | 0 |
| T5 | 2 | 0 |
| Tổng | 13 | 0 |

Các mục trên được tính như sau:

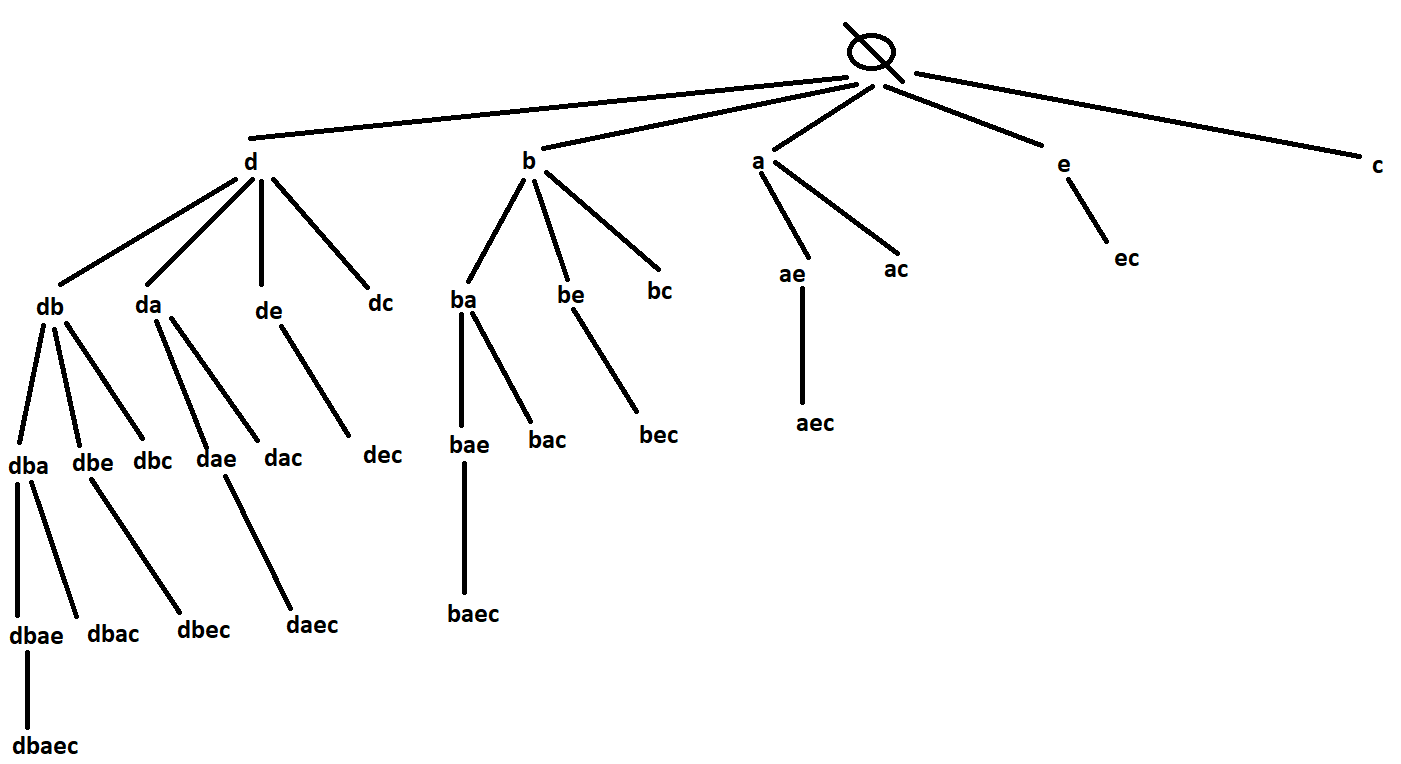
Ví dụ danh mục {D}:

u(D,T1) = 2, ru(D,T1) = u(A,T1) + u(C,T1) = 5+1 = 6

Những phần còn lại được xử lý theo các tương tự, đối với những danh mục ở sau việc tính ru(X,T) sẽ được lấy tổng những phần tử sau nó theo thứ tự từ trước là:

**D < B < A < E < C.**

Sau khi xét danh sách tiện ích của 1 phần tử ta đến xét tiếp danh sách của 2 phần tử để dễ nhìn và hình dung nên nhóm sẽ đưa ra một sơ đồ như sau:



Khi đã xác định được danh sách tiện ích của 1 mục thì ta tiến hành xet danh sách tiện ích của 2 mục. Về tổng quát thì danh sách tiện ích của 2 mục {xy} có thể được xây dựng bằng giao điểm của danh sách tiện ích của {x} và của {y} sau đó ta sẽ đi tính các giá trị Iutil và Rutil của {xy}, các giá trị này sẽ được tính như là cách tính của 1 danh mục nhưng ở đây chúng ta sẽ lấy tổng của {x} và {y}.

Lần lượt hội item d với các phần tử đứng phía sau d trong quan hệ thứ tự **D < B < A < E < C** để tạo ra tập item mới: { D,B }, { D,A },{ D,E },{ D,C }, tính UL cho các tập mới phát sinh này.

{DB}

|  |  |  |
| --- | --- | --- |
| **Tid** | **Iutils** | **Rutils** |
| **T3** | 16 | 9 |
| **T4** | 14 | 6 |
| **Tổng** | 30 | 15 |

{DA}

|  |  |  |
| --- | --- | --- |
| **Tid** | **Iutils** | **Rutils** |
| **T1** | 7 | 1 |
| **T3** | 17 | 4 |
| **Tổng** | 24 | 5 |

{DE}

|  |  |  |
| --- | --- | --- |
| **Tid** | **Iutils** | **Rutils** |
| **T3** | 15 | 1 |
| **T4** | 9 | 3 |
| **Tổng** | 24 | 4 |

{DC}

|  |  |  |
| --- | --- | --- |
| **Tid** | **Iutils** | **Rutils** |
| **T1** | 3 | 0 |
| **T3** | 13 | 0 |
| **T4** | 9 | 0 |
| **Tổng** | 33 | 0 |

Tương tự ta sẽ đi tính toán những danh mục còn lại:

{BA}

|  |  |  |
| --- | --- | --- |
| **Tid** | **Iutils** | **Rutils** |
| **T3** | 9 | 4 |
| **Tổng** | 9 | 4 |

{BE}

|  |  |  |
| --- | --- | --- |
| **Tid** | **Iutils** | **Rutils** |
| **T4** | 11 | 3 |
| **Tổng** | 11 | 3 |

{BC}

|  |  |  |
| --- | --- | --- |
| **Tid** | **Iutils** | **Rutils** |
| **T4** | 11 | 0 |
| **T5** | 6 | 0 |
| **Tổng** | 17 | 0 |

{AE}

|  |  |  |
| --- | --- | --- |
| **Tid** | **Iutils** | **Rutils** |
| **T2** | 16 | 6 |
| **T3** | 8 | 1 |
| **Tổng** | 24 | 7 |

{AC}

|  |  |  |
| --- | --- | --- |
| **Tid** | **Iutils** | **Rutils** |
| **T1** | 6 | 0 |
| **T2** | 16 | 0 |
| **T3** | 6 | 0 |
| **Tổng** | 28 | 0 |

{EC}

|  |  |  |
| --- | --- | --- |
| **Tid** | **Iutils** | **Rutils** |
| **T2** | 12 | 0 |
| **T3** | 4 | 0 |
| **T4** | 6 | 0 |
| **T5** | 5 | 0 |
| **Tổng** | 27 | 0 |

{DBA}

|  |  |  |
| --- | --- | --- |
| **Tid** | **Iutils** | **Rutils** |
| **T3** | 21 | 4 |
| **Tổng** | 21 | 4 |

{DBE}

|  |  |  |
| --- | --- | --- |
| **Tid** | **Iutils** | **Rutils** |
| **T3** | 19 | 1 |
| **T4** | 17 | 3 |
| **Tổng** | 36 | 4 |

{DBC}

|  |  |  |
| --- | --- | --- |
| **Tid** | **Iutils** | **Rutils** |
| **T3** | 17 | 0 |
| **T4** | 17 | 0 |
| **Tổng** | 34 | 0 |

Lần lượt tính các giá trị tổng của iulti và rulti của tập item có 1 phần tử, 2 phần tử và k phần tử (k≥3) để tìm ra các tập tiện ích cao với giá trị minutil >= 40

Đầu tiên ta xét danh mục D: có giá trị tổng iutils = 20, tổng giá trị Rutils = 33, ta có iutils + rutils = 20 + 33 = 53 > 40 nên:

Gán tập exULs = NULL

Lần lượt hội item d với các phần tử đứng phía sau d trong quan hệ thứ tự D < B < A < E < C để tạo ra tập item mới: { D,B }, { D,A },{ D,E },{ D,C }, tính UL cho các tập mới phát sinh này.

UL1({D,B}) = (30,15): ta có iutils + rutils = 30 + 15 = 45 40 đưa vào exULs.

UL2({D,A}) = (24,5): ta có iutils + rutils = 24 + 5 = 29 < 40 không đưa vào exULs.

UL3({D,E}) = (24,4): ta có iutils + rutils = 24 + 4 = 28 < 40 không đưa vào exULs.

UL4(D,C) = (33,0): ): ta có iutils + rutils = 33 + 0 = 33 < 40 không đưa vào exULs.

Do UL1({D,B}) = 45 > 40 nên ta đi xét ta đi xét tiếp các tập {D,B,A}, {D,B,E}, {D,B,C} như sau:

UL5({DBA}) = (21,4): ta có iutils + rutils = 21 + 4 = 25 < 40 không đưa vào exULs.

UL6({DBE}) = (36,4): ta có iutils + rutils = 36 + 4 = 40 40 đưa vào exULs.

UL7({DBC}) = (34,0): ta có iutils + rutils = 34 + 0 = 34 < 40 không đưa vào exULs.

Do UL6(DBE) = 40 minutil nên ta sẽ xét tiếp các tập phát sinh {DBEC} như sau:

|  |  |  |
| --- | --- | --- |
| **Tid** | **Iutils** | **Rutils** |
| **T3** | 20 | 0 |
| **T4** | 20 | 0 |
| **Tổng** | 40 | 0 |

UL7({DBEC) = (40,0): ta có iutils + rutils = 40 + 0 = 40 40 đưa vào exULs.

Kết thúc vì không còn tập phát sinh.

Ta xét danh mục B: có giá trị tổng iutils = 16, tổng giá trị Rutils = 20, ta có iutils + rutils = 16 + 20 = 36 < 40 nên không mở rộng các phần tử sau danh mục B.

Xét danh mục A: có giá trị tổng iutils = 20, tổng giá trị Rutils = 17, ta có iutils + rutils = 20 + 17 = 37 < 40 nên không mở rộng các phần tử sau danh mục A.

Xét danh mục E: có giá trị tổng iutils = 15, tổng giá trị Rutils = 12, ta có iutils + rutils = 15 + 12 = 27 < 40 nên không mở rộng các phần tử sau danh mục E.

Xét danh mục C: có giá trị tổng iutils = 13, tổng giá trị Rutils = 0, ta có iutils + rutils = 13 + 0 = 13 < 40 nên không mở rộng các phần tử sau danh mục C.

Vậy exULs sẽ có các tập sau: {DB}, {DBE}, {DBEC} đây là các mục tiện ích cao mà HUI-Miner khai thác được.

* + 1. **KẾT LUẬN**

Khai thác tập có ích cao là một nhiệm vụ mang tính thử thách trong khai thác mẫu tuần tự, lĩnh vực có nhiều ứng dụng rộng rãi. Thuật toán điển hình là HUIMiner. Thuật toán này sử dụng phương pháp tìm kiếm theo chiều sâu để tìm ra các mẫu và tính toán lợi ích của chúng mà không tốn chi phí cho việc duyệt CSDL. Dù hướng tiếp cận này có hiệu quả, việc khai thác các tập có ích cao vẫn còn tốn kém vì HUI-Miner phải thực hiện thao tác kết các item được tạo ra bằng thủ tục tìm kiếm.

**TÀI LIỆU THAM KHẢO**

**Tiếng Việt**

[1] ThS. Nguyễn Thị Thanh Thủy, ThS. Lâm Thị Họa Mi, Hệ quản trị cơ sở dữ liệu (giáo trình nội bộ), Trường đại học công nghiệp thực phẩm TP.HCM- Năm 2016.

**Website**