10/18/2018

**KHAI PHÁ DỮ LIỆU**

Nhóm 4 – CNTT2-K56

**Thành viên:**

* Đào Văn Cường
* Đỗ Phú Cường
* Nguyễn Văn Quyền
* Nguyễn Văn Viện
* Phan Văn Tiến

# **Lời mở đầu**

Ngày nay, dữ liệu càng nhiều, dẫn đến rất khó cho các nhà phân tích để đưa ra chiến lược hợp lý trong kinh doanh, y học, … Chúng ta cần một thuật toán để phân tích nhằm đưa ra cái nhìn tổng quan về dữ liệu đang được phân tích và có quyết định chính xác hơn.

Làm thế nào một nhân viên sale có thể bán được nhiều hàng hóa hơn, các siêu thị có thể đặt các mặt hàng nhằm kích thích mua hàng cho khách hàng. Chúng ta phải tìm luật kết hợp trong những dữ liệu mà chúng ta có. Vậy có hàng trăm nghìn bản ghi làm thế nào có thể xử lý nhanh mà vẫn chính xác. Sau đây chúng ta sẽ tìm hiểu giải thuật Apriori. Dù khá lâu đời những chúng ta vẫn có thể áp dụng trong thực tế ngày nay.

**MỤC LỤC**

[**Lời mở đầu** 1](#_Toc528181998)

[**I.** **Phương pháp luận kết hợp** 3](#_Toc528181999)

[**1.** **Giới thiệu luật kết hợp** 3](#_Toc528182000)

[**2.** **Phát hiện luật kết hợp** 4](#_Toc528182001)

[**3.** **Chiến lược sinh tập thường xuyên** 6](#_Toc528182002)

[**II.** **Ứng dụng của khai phá dữ liệu** 6](#_Toc528182003)

[**1.** **Hỗ trợ ra quyết định nhập kho trong siêu thị** 6](#_Toc528182004)

[**2.** **Tiếp thị chéo** 7](#_Toc528182005)

[**III.** **Giải thuật Apriori** 9](#_Toc528182006)

[**Lời kết** 12](#_Toc528182007)

[**Tài Liệu Tham Khảo** 12](#_Toc528182008)

1. **Phương pháp luận kết hợp**
2. **Giới thiệu luật kết hợp**

Bài toán phát hiện luật kết hợp (association rule mining): với một tập hợp  
các giao dịch cho trước, cần tìm các luật dự đoán khả năng xuất hiện trong một  
giao dịch cảu các mục (items) này dựa trên việc xuất hiện của các mục khác.



|  |  |
| --- | --- |
| TID | Items |
| 1 | Bread, Milk |
| 2 | Bread, Diaper, Beer, Eggs |
| 3 | Milk, Diaper, Beer, Coke |
| 4 | Bread, Milk, Diaper, Beer |
| 5 | Bread, Milk, Diaper, Coke |

***Các định nghĩa cơ bản:***

* **Tập mục (itemset):** là một tập hợp gồm một hoặc nhiều mục. Tâp mục mức k (k-itemset) có k mục.

*Ví dụ:* 3-itemset là {Milk, Bread, Diaper}.

* **Luật kết hợp:** kí hiệu X -> Y, trong đó X, Y là các tập mục.
* **Tổng số hỗ trợ (support count):** kí hiệu σ : là số lần xuất hiện của một tập mục.

*Ví dụ:* σ ({Milk, Bread, Diaper}) = 2.

* Độ hỗ trợ (support)- kí hiệu s: là tỷ lệ các giao dịch chứa cả X và Y đối với tất cả các giao dịch.

*Ví dụ:* s ({Milk, Diaper, Beer}) = 2/5.

* Độ tin cậy (confidence) – kí hiệu c: là tỷ lệ các giao dịch chứa cả X và Y đối với các giao dịch chứa X.

*Ví dụ:* c ({Milk, Diaper, Beer}) = 2/3.

* + Tập mục thường xuyên (frequent/large itemset): là tập mục mà độ hỗ trợ lớn hơn hoặc bằng một giá trị ngưỡng min-sup.

1. **Phát hiện luật kết hợp**

Với một tập các giao dịch T, mục đích của bài toán phát hiện luật kết hợp là tìm ra tất cả các luật có:

* Độ hỗ trợ s ≥ giá trị ngưỡng **minsup**
* Độ tin cậy ≥ giá trị ngưỡng **minconf**.

Cách tiếp cận vét cạn (Brute-force):

* Liệt kê tất cả các luật kết hợp có thể.
* Tính toán độ hỗ trợ và độ tin cậy cho mỗi luật.
* Loại bỏ đi các luật có độ hỗ trợ nhỏ hơn minsup hoặc có độ tin cậy nhỏ hơn min-conf.

⇒ Phương pháp vét cạn này có chi phí tính toán quá lớn, không áp dụng được trong thực tế.

Xét tập mục: {Milk, Diaper, Beer}

|  |  |
| --- | --- |
| TID | Items |
| 1 | Bread, Milk |
| 2 | Bread, Diaper, Beer, Eggs |
| 3 | Milk, Diaper, Beer, Coke |
| 4 | Bread, Milk, Diaper, Beer |
| 5 | Bread, Milk, Diaper, Coke |

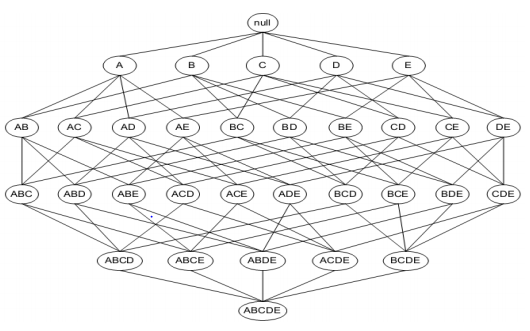


Ta thấy tất cả các luật trên đều là sự phân tách (thành 2 tập con) của cùng tập mục: {Milk, Diaper, Beer}.

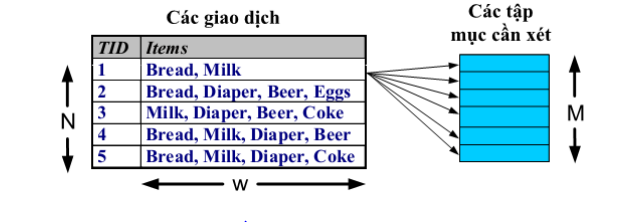
Các luật sinh ra từ cùng một tập mục sẽ có cùng độ hỗ trợ, nhưng có thể khác về độ tin cậy. Do đó, trong quá trình phát hiện luật kết hợp, chúng ta có thể tách riêng 2 yêu cầu về độ hỗ trợ và độ tin cậy. Vây nên quá trình phát hiện luật kết hợp sẽ phần gồm 2 bước (2 giai đoạn) quan trọng:

* Sinh ra các tập mục thường xuyên (frequent/large itemsets): Sinh ra tất cả các tập mục có độ hỗ trợ ≥ **minsup**.
* Sinh ra các luật kết hợp: Từ mỗi tập mục thường xuyên (thu được ở bước  
  trên), sinh ra tất cả các luật có độ tin cậy cao( ≥ **minconf**).  
  Tuy vậy, bước sinh ra các tập mục thường xuyên (bước 1) vẫn có chi phí  
  tính toán quá cao.

Với *d* mục, thì phải xét đến 2d các tập mục có thể.



Với phương pháp vét cạn(Brute-force) để sinh ra các tập mục thường xuyên  
(bước 1):



* Mỗi tập mục trong lược đồ đều được xét.
* Tính độ hỗ trợ của mỗi tập mục, bằng cách duyệt qua tất cả các giao dịch.
* Với mỗi giao dịch, so sánh nó với mỗi tập mục được xét.
* Độ phức tạp ~ O(N.M.w). Nếu M = 2*d* thì độ phức tạp này là quá lớn.

1. **Chiến lược sinh tập thường xuyên**

Dựa vào các phân tích ở mục 1.6.2, ta có các chiến lược:

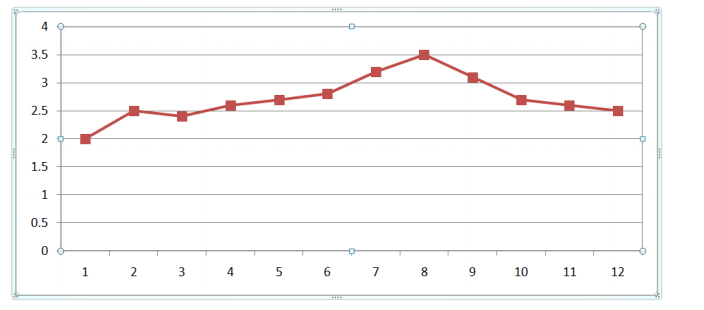
* Giảm bớt số lượng các tập mục cần xét (M): Tìm kiếm (xét) đầy đủ M = 2d. Sau đó, sử dụng các kỹ thuật cắt tỉa để giảm giá trị M.
* Giảm bớt số lượng các giao dịch cần xét (N): Giảm giá trị N, khi kích thước (số lượng các mục) của tập mục tăng lên.
* Giảm bớt số lượng các so sánh (matchings/comparisons) giữa các tập mục và các giao dịch (N.M ): Sử dụng các cấu trúc dữ liệu phù hợp (hiệu quả) để lưu các tập mục cần xét hoặc các giao dịch. Không cần phải so sánh mỗi tập mục với mỗi giao dịch.

1. **Ứng dụng của khai phá dữ liệu**
2. **Hỗ trợ ra quyết định nhập kho trong siêu thị**

**Giới thiệu về bài toán:**

|  |  |
| --- | --- |
| Tháng | Sản lượng |
| 1 | 2 |
| 2 | 2.5 |
| 3 | 2.4 |
| 4 | 2.6 |
| 5 | 2.7 |
| 6 | 2.8 |
| 7 | 3.2 |
| 8 | 3.5 |
| 9 | 3.1 |
| 10 | 2.7 |
| 11 | 2.6 |
| 12 | 2.5 |





1. **Tiếp thị chéo**

**Giới thiệu về bài toán:**

Trong một cửa hàng bán lẻ thì người quản lý có rất nhiều cách thức trong việc sắp xếp thứ tự vị trí của các sản phẩm mà họ đang kinh doanh. Trong thực tế thì khách hàng khi mua một sản phẩm A thì thường hay có xu hướng tiếp tục mua tiếp các sản phẩm B, C, D….. có liên quan đến sản phẩm A. Do đó người quản lý phải tìm hiểu về giỏ hàng mà khách hàng thường hay thực hiện trong các giao dịch để rút ra quy luật để từ đó tiến hành sắp xếp lại các mặt hàng thường được mua cùng nhau để đặt chúng cạnh nhau nhằm giúp giảm công sức đi lại cho khách hàng, gợi ý mua hàng để tăng doanh số bán hàng. Để có thể thực hiện được mục tiêu như trên thì ta có thể áp dụng khai phá luật kết hợp để rút ra các quy luật đó.

***Ví dụ:*** Cửa hàng bán lẻ trích ra một số giao dịch mà khách hàng đã thực hiện trong lịch sử giao dịch trước đây của cửa hàng.

|  |  |
| --- | --- |
| TID | List of item\_IDs |
| T100 | I1, I2, I5 |
| T200 | I2, I4 |
| T300 | I2, I3 |
| T400 | I1, I2, I4 |
| T500 | I1, I3 |
| T600 | I2, I3 |
| T700 | I1, I3 |
| T800 | I1, I2, I3, I5 |
| T900 | I1, I2, I3 |

Sau khi tiến hành áp dụng một giải thuật khai phá luật kết hợp (vd:Apriori) thì ta được bảng Frequent Itemset như sau:

|  |  |
| --- | --- |
| Itemset | Support Count |
| {I1, I2, I3} | 2 |
| {I1, I2, I5} | 2 |

Từ bảng này thì người quản lý của cửa hàng bán lẻ sẽ yêu cầu nhân viên của  
mình tiến hành đặt các sản phẩm có trong Itemset trong cùng một hàng gần nhau (cùng một gian hàng) để giúp người mua hàng và kích thích người tiêu dùng.

1. **Giải thuật Apriori**

Quá trình sinh ra luật kết hợp chia làm hai bước. Bước đầu tiên là sinh ra các tập thường xuyên. Bước thứ hai sinh ra các luật kết hợp. Ở mục 1.6.3 ta thấy được bước thứ nhất của quá trình rất phức tạp. Giải thuật Apriori là một phương pháp làm giảm độ phức tạp ở bước này.

Nguyên tắc của giải thuật Apriori – Loại bỏ dựa trên độ hỗ trợ:

* Nếu một tập mục là thường xuyên, thì tất cả các tập con (subsets) của nó đều là các tập mục thường xuyên.
* Nếu một tập mục là không thường xuyên (not frequent) thì tất cả các tập cha (supersets) của nó đều là các tập mục không thường xuyên. Nguyên tắc của giải thuật Apriori dựa trên đặc tính không đơn điệu (antimonotone) của độ hỗ trợ:

∀ X, Y : (X ⊆ Y) ⇒ s(X) ≥ s(Y)

***Ví dụ:*** Loại bỏ dựa trên độ hỗ trợ *minsup = 3*

|  |  |
| --- | --- |
| Itemset | Count |
| {Bread,Milk,Diaper} | 3 |
| **Nếu xét các tập mục có thể:**  **6+21+14=41** | |
| Với cơ chế loại bỏ độ hỗ trợ:  6+6+1=13 | |

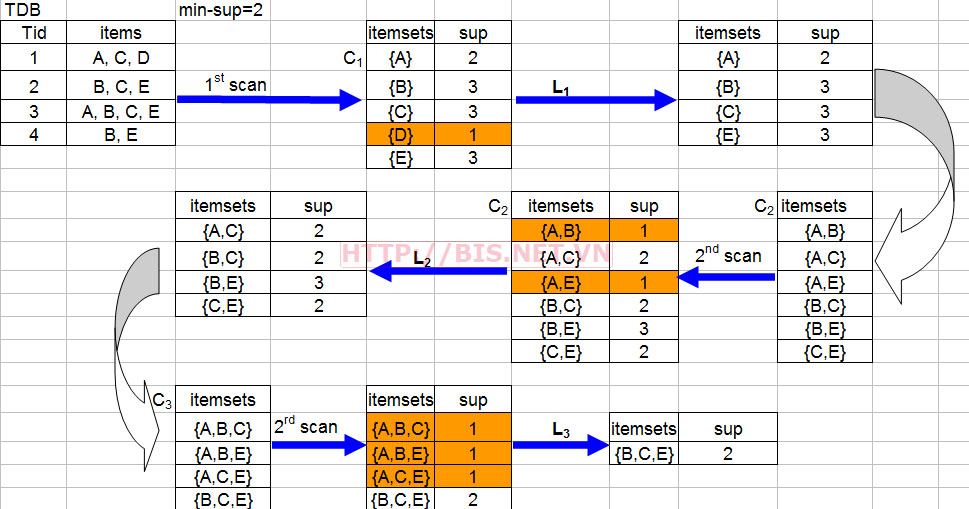
|  |  |
| --- | --- |
| Item | Count |
| {Bread,Milk} | 3 |
| {Bread,Beer} | 2 |
| {Bread,Diaper} | 3 |
| {Milk,Beer} | 2 |
| {Milk,Diaper} | 3 |
| {Beer,Diaper} | 3 |

|  |  |
| --- | --- |
| Item | Count |
| Bread | 4 |
| Coke | 2 |
| Milk | 4 |
| Beer | 3 |
| Diaper | 4 |
| Eggs | 1 |

***Các bước thực hiện:***

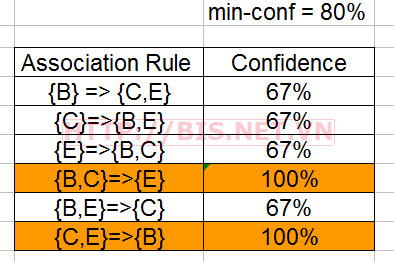
* Sinh ra tất cả các tập mục thường xuyên mức 1(frequent 1-itemsets)
* Gán k =1
* Lặp lại, cho đến khi không có thêm bất kỳ tập mục thường xuyên nào  
  mới.
  + Từ các tập mục thường xuyên mức k, sinh ra các tập mục mức (k+1)  
    cần xét.
  + Loại bỏ các tập mục mức k+1 chứa các tập con là các tập mục không  
    thường xuyên mức k.
  + Tính độ hỗ trợ của các tập mục mức k+1, bằng cách duyệt qua tất cả  
    các giao dịch.
  + Loại bỏ các tập mục không thường xuyên mức k+1 .
  + Thu được các tập mục thường xuyên mức k+1

Ví dụ:



* Với mỗi tập mục thường (I) thu được, sinh ra tất cả các tập con (B)  
  không rỗng
* Với mỗi tập B, sinh ra các luật kết hợp: B (I-B)
* Với mỗi luật kết hợp, duyệt qua tất cả các giao dịch. Chọn các luật có độ tin cậy(c) ≥ minconf.

Ví dụ:



# **Lời kết**

Như vậy chúng ta đã tìm hiểu xong giải thuật Apriori. Tuy giải thuật còn nhiều nhiều hạn chế. Xong chúng ta có thể áp dụng chúng vào thực tế để tìm ra những luật kết hợp để áp dụng vào thực tế như: bán hàng, sắp xếp các mặt hàng, …

Do năng lực còn hạn chế nên nhóm còn nhiều thiếu sót và sơ sài. Mong thầy thấu hiểu.

Song, nhóm em cũng cảm ơn những lời góp ý nhiệt tình từ thầy và các bạn để nhóm em có thể tiến bộ hơn.

# **Tài Liệu Tham Khảo**

* <http://myweb.sabanciuniv.edu/rdehkharghani/files/2016/02/The-Morgan-Kaufmann-Series-in-Data-Management-Systems-Jiawei-Han-Micheline-Kamber-Jian-Pei-Data-Mining.-Concepts-and-Techniques-3rd-Edition-Morgan-Kaufmann-2011.pdf>
* <http://www.cit.ctu.edu.vn/~dtnghi/dataminingR/ass.pdf>
* <https://en.wikipedia.org/wiki/Apriori_algorithm>
* <https://en.wikipedia.org/wiki/Apriori_algorithm>
* <http://bis.net.vn/forums/t/389.aspx>