Bài 5: KHAI THÁC MẪU KẾT HỢP (TT)

I. Mục tiêu:

Sau khi thực hành xong, sinh viên nắm được:

- Thuật toán Vertical Apriori.

II. Tóm tắt lý thuyết:

Thuật toán Apriori áp dụng cho một tập dữ liệu dạng nằm ngang (horizontal) còn thuật toán vertical Apriori áp dụng cho một tập dữ liệu dạng thẳng đứng (vertical).

Thuật toán Vertical Apriori được phát biểu như sau:

Algorithm *VerticalApriori*(Transactions: $\mathcal{T}$, Minimum Support: *minsup*)
begin
  $k = 1$;
  $\mathcal{F}_1$ = { All Frequent 1-itemsets };
  Construct vertical *tid* lists of each frequent item;
  while $\mathcal{F}_k$ is not empty **do begin**
    Generate $\mathcal{C}_{k+1}$ by joining itemset-pairs in $\mathcal{F}_k$;
    Prune itemsets from $\mathcal{C}_{k+1}$ that violate downward closure;
    Generate *tid* list of each candidate itemset in $\mathcal{C}_{k+1}$ by intersecting
      tid lists of the itemset-pair in $\mathcal{F}_k$ that was used to create it;
    Determine supports of itemsets in $\mathcal{C}_{k+1}$ using lengths of their *tid* lists;
    $\mathcal{F}_{k+1}$= Frequent itemsets of $\mathcal{C}_{k+1}$ together with their *tid* lists;
    $k = k + 1$;
  end;
  return$(\cup_{i=1}^{k} \mathcal{F}_i)$;
end

Ví dụ: cho tập dữ liệu giao dịch được lưu trữ dưới dạng nằm ngang với minsup = 4 (60%), min confidence = 80%

TID	Itemset
1	Wine, Chips, Break, Milk
2	Wine, Break, Milk
3	Break, Milk
4	Chips
5	Wine, Chips, Break, Milk
6	Wine, Chips, Milk

Chuyển dữ liệu từ dạng nằm ngang thành dạng thẳng đứng

Items	TID list
Wine	1,2,5,6
Break	1,2,3,5
Chips	1, 4, 5, 6
Milk	1, 2, 3, 5, 6

Do $minsup = 4$ nên tất cả các item xuất hiện ít hơn 4 lần trong giao dịch sẽ được loại trừ khỏi tập dữ liệu

Items	TID list	frequency
Wine	1,2,5,6	4
Break	1,2,3,5	4
Chips	1, 4, 5, 6	4
Milk	1, 2, 3, 5, 6	5

Tiếp theo ta tạo 1 danh sách chứa 2-item

Items
Wine, Break
Wine, Chips
Wine, Milk
Break, Chips
Break, Milk
Chips, Milk

Ta kết hợp tất cả item với TID tương ứng

Itemset	TID list	frequency
Wine, Break	1, 2, 5	3
Wine, Chips	1,5,6	3
Wine, Milk	1,2,5,6	4
Break, Chips	1,5	2
Break, Milk	1,2,3,5,6	5
Chips, Milk	1,5,6	3

Ta bỏ các 2-item có giá trị support nhỏ hơn $minsup = 4$:

Itemset	TID list	frequency
Wine, Milk	1,2,5,6	4
Break, Milk	1,2,3,5,6	5

Tạo 3- item giao dich và nó có giá trị support nhỏ hơn giá trị của $minsup = 4$

Items	TID list	frequency
Break, Milk, Wine	1,2,5	3

nên ta sẽ khởi tạo luật kết hợp dựa vào bước trước đó

Itemset	TID list	frequency
Wine, Milk	1,2,5,6	4
Break, Milk	1,2,3,5,6	5

Phát sinh các luật:

$$Wine \rightarrow Milk \text{ có } conf(Wine \rightarrow Milk) = \frac{support(Wine, Milk)}{support(Wine)} = 4/4 = 100\%$$

$$Milk \rightarrow Wine \text{ có } conf(Milk \rightarrow Wine) = \frac{support(Wine, Milk)}{support(Milk)} = 4/5 = 80\%$$

$$Bread \rightarrow Milk \text{ có } conf(Bread \rightarrow Milk) = \frac{support(Bread, Milk)}{support(Bread)} = 4/4 = 100\%$$

$$Milk \rightarrow Bread \text{ có } conf(Milk \rightarrow Bread) = \frac{support(Milk, Bread)}{support(Milk)} = 4/5 = 80\%$$

Vậy ta có 2 luật sau:

$$Wine \rightarrow Milk \text{ có } conf(Wine \rightarrow Milk) = 100\%$$

$$Bread \rightarrow Milk \text{ có } conf(Bread \rightarrow Milk) == 100\%$$

III. Nội dung thực hành:

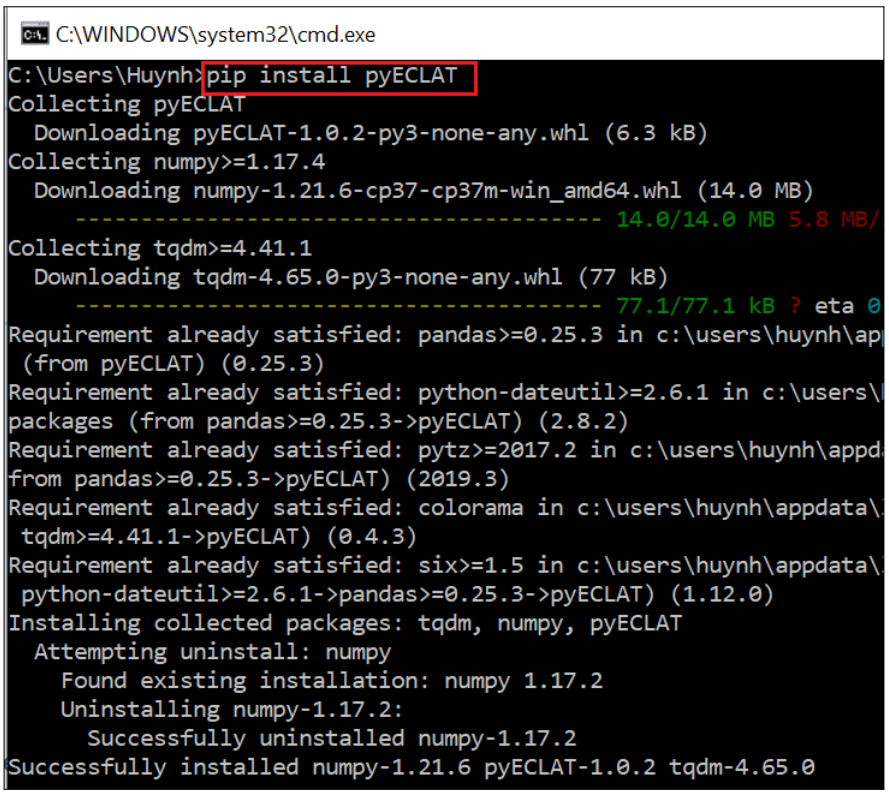
Cho CSDL với các giao dịch sau:

TID	Itemset
1	Wine, Chips, Bread, Butter, Milk, Apple
2	Wine, Bread, Butter, Milk
3	Bread, Butter, Milk
4	Chips, Apple
5	Wine, Chips, Bread, Butter, Milk, Apple
6	Wine, Chips, Milk
7	Wine, Chips, Bread, Butter, Apple
8	Wine, Chips, Milk
9	Wine, Bread, Apple
10	Wine, Bread, Butter, Milk
11	Chips, Bread, Butter, Apple
12	Wine, Butter, Milk, Apple
13	Wine, Chips, Bread, Butter, Milk
14	Wine, Bread, Milk, Apple
15	Wine, Bread, Butter, Milk, Apple
16	Wine, Chips, Bread, Butter, Milk, Apple
17	Chips, Bread, Butter, Milk, Apple
18	Chips, Butter, Milk, Apple
19	Wine, Chips, Bread, Butter, Milk, Apple
20	Wine, Bread, Butter, Milk, Apple
21	Wine, Chips, Bread, Milk, Apple
22	Chips

với $minsup = 60\%$.

1. Sử dụng thư viện

- Cài đặt pyECLAT (pip install pyECLAT):



- Tạo file data.csv (sử dụng lại data.csv của Bài 4) và đọc dữ liệu từ file.

```
import numpy as np
import pandas as pd
from pyECLAT import ECLAT

dataframe = pd.read_csv('D:\\Huynh\\DataMining_Lab\\data\\tuan5\\data.csv', header=None)
print(dataframe)
```

	0	1	2	3	4	5
0	Wine	Chips	Bread	Butter	Milk	Apple
1	Wine	NaN	Bread	Butter	Milk	NaN
2	NaN	NaN	Bread	Butter	Milk	NaN
3	NaN	Chips	NaN	NaN	NaN	Apple
4	Wine	Chips	Bread	Butter	Milk	Apple
5	Wine	Chips	NaN	NaN	Milk	NaN
6	Wine	Chips	Bread	Butter	NaN	Apple
7	Wine	Chips	NaN	NaN	Milk	NaN
8	Wine	NaN	Bread	NaN	NaN	Apple
9	Wine	NaN	Bread	Butter	Milk	NaN
10	NaN	Chips	Bread	Butter	NaN	Apple
11	Wine	NaN	NaN	Butter	Milk	Apple
12	Wine	Chips	Bread	Butter	Milk	NaN
13	Wine	NaN	Bread	NaN	Milk	Apple
14	Wine	NaN	Bread	Butter	Milk	Apple
15	Wine	Chips	Bread	Butter	Milk	Apple
16	NaN	Chips	Bread	Butter	Milk	Apple
17	NaN	Chips	NaN	Butter	Milk	Apple
18	Wine	Chips	Bread	Butter	Milk	Apple
19	Wine	NaN	Bread	Butter	Milk	Apple
20	Wine	Chips	Bread	NaN	Milk	Apple
21	NaN	Chips	NaN	NaN	NaN	NaN

- Chuyển dữ liệu thành lớp ECLAT và khởi tạo DataFrame nhị phân

```
eclat_instance = ECLAT(data=dataframe, verbose=True)
print(eclat_instance.df_bin)
```

	Butter	Chips	Wine	Bread	Apple	Milk
0	1	1	1	1	1	1
1	1	0	1	1	0	1
2	1	0	0	1	0	1
3	0	1	0	0	1	0
4	1	1	1	1	1	1
5	0	1	1	0	0	1
6	1	1	1	1	1	0
7	0	1	1	0	0	1
8	0	0	1	1	1	0
9	1	0	1	1	0	1
10	1	1	0	1	1	0
11	1	0	1	0	1	1
12	1	1	1	1	0	1
13	0	0	1	1	1	1
14	1	0	1	1	1	1
15	1	1	1	1	1	1
16	1	1	0	1	1	1
17	1	1	0	0	1	1
18	1	1	1	1	1	1
19	1	0	1	1	1	1
20	0	1	1	1	1	1
21	0	1	0	0	0	0

- Khởi tạo các luật kết hợp

```
# count items in each row
items_per_transaction = eclat_instance.df_bin.astype(int).sum(axis=1)
# the item shoud appear at least at 5% of transactions
min_support = 0.6
# start from transactions containing at least 2 items
min_combination = 2
# up to maximum items per transaction
max_combination = max(items_per_transaction)
rule_indices, rule_supports = eclat_instance.fit(min_support=min_support,
                                                 min_combination=min_combination,
                                                 max_combination=max_combination,
                                                 separator=' & ',
                                                 verbose=True)

result = pd.DataFrame(rule_supports.items(),columns=['Item', 'Support'])
result1=result.sort_values(by=['Support'], ascending=False)
print(result1)
```

	Item	Support
0	Wine & Milk	0.636364

2. Yêu cầu:

- Cài đặt lại thuật toán Vertical Apriori.

- Viết file báo cáo trình bày tóm tắt lại phần code do em tự viết và so sánh kết quả với hàm có sẵn trong thư viện.