Kai Wong

UID: 704451679

UCLA CS 145, Fall 2018, Homework 5

**1. Frequent Pattern Mining for Set Data**

**1a. Find all the frequent patterns using Apriori Algorithm. Details of the procedure are expected.**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **C1** | **Itemset** | **sup** | **L1** | **Itemset** | **sup** |
| {a} | 6 | {a} | 6 |
| {b} | 8 | {b} | 8 |
| {c} | 6 | {c} | 6 |
| {d} | 4 | {d} | 4 |
| {e} | 2 | {e} | 2 |
| {i} | 1 |
| {j} | 1 |
| {k} | 1 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **C2** | **Itemset** | **sup** | **L2** | **Itemset** | **sup** |
| {a,b} | 4 | {a,b} | 4 |
| {a,c} | 4 | {a,c} | 4 |
| {a,d} | 2 | {a,d} | 2 |
| {a,e} | 2 | {a,e} | 2 |
| {b,c} | 4 | {b,c} | 4 |
| {b,d} | 4 | {b,d} | 4 |
| {b,e} | 2 | {b,e} | 2 |
| {c,d} | 1 |
| {c,e} | 1 |
| {d,e} | 0 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **C3** | **Itemset** | **sup** | **L3** | **Itemset** | **sup** |
| {a,b,c} | 2 | {a,b,c} | 2 |
| {a,b,d} | 2 | {a,b,d} | 2 |
| {a,b,e} | 2 | {a,b,e} | 2 |
| {a,c,d} | 1 |
| {a,c,e} | 1 |
| {a,d,e} | 0 |
| {b,c,d} | 1 |
| {b,c,e} | 1 |
| {b,d,e} | 0 |

|  |  |  |
| --- | --- | --- |
| **C4** | **Itemset** | **sup** |
| {a,b,c,d} | 1 |
| {a,b,c,e} | 1 |
| {a,b,d,e} | 0 |

**L4 :**

**1b. Construct and draw the FP-tree of the transaction database.**

Frequent 1-itemset:

|  |  |
| --- | --- |
| **Item** | **Frequency** |
| {a} | 6 |
| {b} | 8 |
| {c} | 6 |
| {d} | 4 |
| {e} | 2 |
| {i} | 1 |
| {j} | 1 |
| {k} | 1 |

**FP-Tree**

{ }

a : 4

b : 8

a : 2

Sort Frequent 1-itemset, prune for min\_support = 2:

|  |  |
| --- | --- |
| **Item** | **Frequency** |
| {b} | 8 |
| {a} | 6 |
| {c} | 6 |
| {d} | 4 |
| {e} | 2 |

c : 2

d : 2

c : 2

Scan database again:

d : 1

e : 1

e : 1

d : 1

c : 2

|  |  |  |
| --- | --- | --- |
| TID | Items | Frequent Items (Ordered) |
| 1 | {b,c,j} | {b,c} |
| 2 | {a,b,d} | {b,a,d} |
| 3 | {a,c} | {a,c} |
| 4 | {b,d} | {b,d} |
| 5 | {a,b,c,e} | {b,a,c,e} |
| 6 | {b,c,k} | {b,c} |
| 7 | {a,c} | {a,c} |
| 8 | {a,b,e,i} | {b,a,e} |
| 9 | {b,d} | {b,d} |
| 10 | {a,b,c,d} | {b,a,c,d} |

**1c. For the item d, show its conditional pattern base and conditional FP-tree.**

Conditional pattern base of item **d: {b:2, bac: 1, ba: 1}**

Using min\_support = 2, **b:2, bac: 1, ba: 1**  **b:4, ba:2**

**d-conditional FP-tree:**

a : 2

{ }

b : 4

**1d. Find frequent patterns based on d’s conditional FP-tree.**

**d-conditional FP-tree: ad-conditional FP-tree:**

{ }

b : 2

{ }

a : 2

b : 4

**Frequent patterns with item d:**

d: 4

bd: 4, ad: 2

bad: 2

**2. Apriori for Yelp**

**Command output:**

C:\Users\kyles\Desktop\CS145\hw5

λ python2 apriori.py

min\_support: 50 min\_conf: 0.25

item: "Wicked Spoon","Holsteins Shakes & Buns" , 51.000

item: "Wicked Spoon","Secret Pizza" , 52.000

item: "Wicked Spoon","Earl of Sandwich" , 52.000

item: "The Cosmopolitan of Las Vegas","Wicked Spoon" , 54.000

item: "Mon Ami Gabi","Wicked Spoon" , 57.000

item: "Bacchanal Buffet","Wicked Spoon" , 63.000

------------------------ RULES:

Rule: "Secret Pizza" ==> "Wicked Spoon" , 0.256

Rule: "The Cosmopolitan of Las Vegas" ==> "Wicked Spoon" , 0.277

Rule: "Holsteins Shakes & Buns" ==> "Wicked Spoon" , 0.315

66.7090001106 sec

**What patterns and rules do you see?**

I got the following frequent itemsets:

{“Wicked Spoon”, “Holsteins Shakes & Buns”}, {“Wicked Spoon”, “Secret Pizza”}, {“Wicked Spoon”, “Earl of Sandwich”}, {“The Cosmopolitan of Las Vegas”, “Wicked Spoon”}, {“Mon Ami Gabi”, “Wicked Spoon”} and {“Bacchanal Buffet”, “Wicked Spoon”}

I got the following association rules:

“Secret Pizza” 🡺 “Wicked Spoon” with confidence 0.256

“The Cosmopolitan of Las Vegas” 🡺 “Wicked Spoon” with confidence 0.277

“Holsteins Shakes & Buns” 🡺 “Wicked Spoon” with confidence 0.315

**Where are these businesses located?**

The businesses are restaurants located at the Las Vegas strip, specifically near or in the hotel “The Cosmopolitan of Las Vegas”. The “Wicked Spoon” which appears in all the frequent itemsets found is a popular high-end buffet located in “The Cosmopolitan of Las Vegas”.

**What do these results mean?**

These results indicate that visitors of the restaurant “Wicked Spoon” frequently also visit some of the other popular restaurants in its vicinity. This may be because most of these visitors are guests of “The Cosmopolitan of Las Vegas” based on its relatively strong association rule with the “Wicked Spoon”.

**3. Correlation Analysis**

**3a. Calculate the confidence, lift, and all\_confidence between buying beer and buying nuts.**

**Confidence:**

*buys beer => buys nuts* 150/500 = 30.0%

*buys beer => not buys nuts* 350/500 = 70.0%

*not buys beer => buys nuts* 700/9500 = 7.4%

*not buys beer => not buys nuts* 8800/9500 = 92.6%

**Lift:**

*lift(buys beer, buys nuts)* = (150/10000) / [(500/10000) \* (850/10000)] = 3.53

*lift(buys beer, not buys nuts)* = (350/10000) / [(500/10000) \* (9150/10000)] = 0.77

*lift(not buys beer, buys nuts)* = (700/10000) / [(9500/10000) \* (850/10000)] = 0.87

*lift(not buys beer, not buys nuts)* = (8800/10000) / [(9500/10000) \* (9150/10000)] = 1.01

**All\_confidence:**

*all\_conf(buys beer, buys nuts)* = min{150/500 = 30.0%, 150/850 = 17.6%} = 17.6%

*all\_conf(buys beer, not buys nuts)* = min{350/500 = 70.0%, 350/9150 = 3.8%} = 3.8%

*all\_conf(not buys beer, buys nuts)* = min{700/9500 = 7.4%, 700/850 = 82.4%} = 7.4%

*all\_conf(not buys beer, not buys nuts)* = min{8800/9500 = 92.6%, 8800/9150 = 96.2%} = 92.6%

**3b. What are your conclusions of the relationship between buying beer and buying nuts, based on the above measures?**

We can conclude that people are likely to buy beer and nuts together. From the data, there is relatively high confidence for both *buys beer => buys nuts* and the inverse *not buys beer => not buys nuts*. However, we also see that there is high confidence for *buys beer => not buys nuts* at 70% but this could be a misleadingly strong association rule since the overall probability of people who do not buy nuts is 9150/10000 = 91.5% >> 70%.

This conclusion is further supported by the lift values we calculated, where only *lift(buys beer, buys nuts)* and *lift(not buys beer, not buys nuts)* have values > 1 which indicate positive correlation between buying beer and buying nuts as well as the inverse (not buying beer and not buying nuts). In addition, the all\_confidence values calculated also support this with the two highest values coming from *all\_conf(buys beer, buys nuts)* and *all\_conf(not buys beer, not buys nuts)* at 17.6% and 92.6% respectively.

**4. Sequential Pattern Mining (GSP Algorithm)**

**4a. For a sequence s = <ab(cd)(ef)>, how many events or elements does it contain? What is the length of s? How many non-empty subsequences does s contain?**

*Number of events/elements = 4*

*Length of s = 6*

Let k be the length of a subsequence, then we have:

|  |  |
| --- | --- |
| k | Number of length-k subsequences |
| 1 | 6 |
| 2 | 15 |
| 3 | 20 |
| 4 | 15 |
| 5 |  |
| 6 | 1 |

Number of non-empty subsequences = 63

**4b. Suppose we have L3 = {<(ac)e>, <b(cd)>, <bce>, <a(cd)>, <(ab)d>, <(ab)c>} as the frequent 3-sequences, write down all the candidate 4-sequences C4 with the details of the join and pruning steps.**

Join step:

<(ac)e> can join with no other 3-sequence

<b(cd)> can join with no other 3-sequence

<bce> can join with no other 3-sequence

<a(cd)> can join with no other 3-sequence

<(ab)d> can join with no other 3-sequence

<(ab)c> join <b(cd)> = <(ab)(cd)>

<(ab)c> join <bce> = <(ab)ce>

Hence, we get *C4 = {<(ab)(cd)>, <(ab)ce>}*

Prune step:

Length 3 subsequences of <(ab)(cd)> = <(ab)c>, <(ab)d>, <a(cd)>, <b(cd)>

All subsequences are in L3.

Length 3 subsequences of <(ab)ce> = <(ab)c>, <(ab)e>, <ace>, <bce>

<(ab)e> is not in L3, so we prune <(ab)ce> from C4.

Hence, L4 = {<(ab)(cd)>}.