Mining both Positive and Negative Association Rules

Xindong Wu (*), Chengqi Zhang (+), and Shichao Zhang (+)

(*) University of Vermont, USA
(+) University of Technology Sydney, Australia

xwu@cs.uvm.edu
Outline

- Negative association rules: examples
- Frequent vs infrequent itemsets
- Defining negative association rules
- Procedure AllItemsOfInterest
- Extracting positive and negative rules
- Algorithm PositiveAndNegativeAssociations
- Some Experimental Results
- Related Work
Negative Association Rules

- E.g. 1: A => B, E => F, where to put C and D?
  (what if A => ~C)

- E.g. 2:
  - t and c: frequent
  - t U c: infrequent
  - support(t U ~c) = support(t) – support(t U c) can be high
  - How about t => ~c? (hint on how to find a negative association rule)
Negative Association Rules

- Exceptional patterns, aka exceptions of rules, surprising rules.

- Eg.
  - Normal rule: Birds(x) => fly(x)
  - Exceptional rule: Birds(x), penguin(x) => ¬fly(x).
Negative Association Rules

Interesting facts: A=>B is a valid rule does not imply ~B=>~A is a valid rule:

- Consider the following database, 
  \{(A,B,C),(A,B),(A,D),(B,C)\},
  
  \[ \text{supp}(A=>B)=\frac{1}{2} > \text{ms}, \ \text{conf}(A=>B)=\frac{2}{3} > \text{mc} \]
  
  \[ \text{supp}(\sim B=>\sim A)=\text{supp}(\sim B U \sim A)=\text{supp}(\sim B)-\text{supp}(A U \sim B)=1-\text{supp}(B)-(\text{supp}(A)-\text{supp}(A U B)) \]
  
  \[ =1-\text{supp}(A)-\text{supp}(B)+\text{supp}(A U B)=1-\frac{3}{4}-\frac{3}{4}+\frac{1}{2}=0 \]
Negative Association Rules

- Infrequent itemsets for negative association rules.
- Negative association rules of form \( A \Rightarrow \sim B \) means \( \text{supp}(A \cup \sim B) \geq ms. \) \( \text{supp}(A \cup B) = \text{supp}(A) - \text{supp}(A \cup \sim B). \)
- For most cases, \( \text{supp}(A) < 2 \times ms. \)
- Therefore, \( \text{supp}(A \cup B) < ms, \) which means \( A \cup B \) is an infrequent itemset, and to find negative association rules, we need to find infrequent itemsets first.
Negative Association Rules

- Generalized negative association rules: a rule that contains a negation of an item. An example: AU\sim BU\sim CUD => EU\sim F.
- This is very difficult because of exponential growth of itemsets.
- Narrow down to the following three cases: A=>\sim B, \sim A=>B, \sim A=>\sim B.
Negative Association Rules

- Still Difficult: exponential growth of infrequent itemsets
  - TD={(A,B,D);(B,C,D);(B,D);(B,C,D,E);(A,B,D,F)}
  - Such a simple database contains 49 infrequent itemsets.
Negative Association Rules

The main challenge:

- how to effectively search for interesting itemsets
- how to effectively identify negative association rules of interest
Frequent vs Infrequent Itemsets

- A frequent itemset I: $support(I) \geq \text{minsupp}$
- An infrequent itemset J: $support(J) < \text{minsupp}$
- How many possible itemsets (m baskets, n items)?
  - $C(m,n)2^m$ (an expensive search process!)
Define Negative Association Rules

- Positive association rules
  - $X \cap Y = \emptyset$
  - $\text{Supp}(X \cup Y) \geq \text{minsupp}$
  - $\frac{\text{Supp}(X \cup Y)}{\text{supp}(X)} \geq \text{minconf}$
Define Negative Association Rules

- Pruning strategy.
  - A => B is of no interest if supp(AUB) \approx supp(A) \times supp(B).
  - It indicates A and B are independent.

- Define a measure interestingness
  - Interest(X, Y) = |supp(XUY) - supp(X) \times supp(Y)|, a threshold mi.
  - An itemset that satisfies the above measure is called a potentially interesting itemset.
Define Negative Association Rules

- Integrating interest measure to support confidence measure (positive association rules)

\[ \text{fi}(I) = \text{supp}(I) \geq ms \land \exists X, Y : X \cup Y = I \land \text{fi}(Y, X) \]

where

\[ \text{fi}(X, Y) = X \cap Y = \emptyset \land \]

\[ f(X, Y, ms, mc, mi) = 1 \]

\[ f(X, Y, ms, mc, mi) = \frac{\text{supp}(X \cup Y) + \text{conf}(X \Rightarrow Y) + \text{interest}(X, Y) - (ms + mc + mi) + 1}{|\text{supp}(X \cup Y) - ms| + |\text{conf}(X \Rightarrow Y) - mc| + |\text{interest}(X, Y) - mi| + 1} \]
Define Negative Association Rules

- Pruning strategy for negative association rules.
  - Eg. Supp(A)>ms, supp(B)<ms and freq(B)=1 in a large database.
  - A=>~B is valid because supp(A)>ms, supp(B) ≈ 0, supp(AU~B) ≈ supp(A)>ms, conf(A=>~B)=supp(AU~B)/supp(A) ≈ 1.
Define Negative Association Rules

- Two cases:
  - If both A and B are frequent, $A \cup B$ is infrequent, is $A \Rightarrow \sim B$ a valid rule?
  - If A is frequent, B is infrequent, is $A \Rightarrow \sim B$ a valid rule? Maybe, but not of our interest.

- **Heuristic**: Only if both A and B are frequent, will $A \Rightarrow \sim B$ be considered.
Define Negative Association Rules

- An interesting negative association rule is defined as follows:
  - \( A \cap B = \emptyset \)
  - \( \text{Supp}(A) \geq \text{minsupp}, \text{supp}(B) > \text{minsupp}, \) and
    \( \text{supp}(A \cup \sim B) \geq \text{minsupp} \)
  - \( \text{Supp}(A \cup \sim B)/\text{supp}(A) \geq \text{minconf} \)
Define Negative Association Rules

- E.g. suppose we have a market basket database, c means coffee, t means tea.
  - Supp(c)=0.6, supp(t)=0.4, supp(tUc)=0.05 and mc =0.52.
  - Supp(tU¬c)=supp(t)-supp(tUc)=0.4-0.05=0.35.
  - Conf(tU¬c)=supp(tU¬c)/supp(t)=0.875>mc
- We have a valid rule t=>¬c.
Define Negative Association Rules

- If $\text{supp}(X) \geq ms$ and $\text{supp}(Y) \geq ms$, the rule $X \Rightarrow \sim Y$ is of potential interest. $XUY$ is called a potentially interesting itemset.

- The pruning strategy ensures we can use an Apriori like algorithm. Generate infrequent $k$ itemsets from frequent $k-1$ itemsets.
Define Negative Association Rules

- Integrating the pruning strategy to the support confidence framework.

\[ \text{Upis}(J) = \text{supp}(J) < ms \land \]
\[ \exists X, Y : X \cup Y = J \land \]
\[ \text{Upis}(X, Y) \]

where

\[ \text{Upis}(X, Y) = X \cap Y = \emptyset \land \]
\[ g(X, \neg Y, ms, mc, mi) = 2 \]
\[ g(X, \neg Y, ms, mc, mi) = f(X, \neg Y, ms, mc, mi) \]
\[ + \frac{\text{supp}(X) + \text{supp}(Y) - 2ms + 1}{|\text{supp}(X) - ms| + |\text{supp}(Y) - ms| + 1} \]
Procedure AllItemsOfInterest

Input: D (a database); minsupp; mininterest

Output: PL (frequent itemsets); NL (infrequent itemsets)

1. let PL ← ∅; NL ← ∅;
2. let $L_1$ ← {frequent 1-itemsets}; PL ← PL ∪ $L_1$;
3. for $(k = 2; (L_{k-1} \neq \emptyset); k ++)$ do
   begin //Generate all possible frequent and infrequent k-itemsets of interest in D.
   3.1 let $Tem_k$ ← \{$x_1, \ldots, x_{k-2}, x_{k-1}, x_k$ | $x_1, \ldots, x_{k-2}, x_{k-1}$ ∈ $L_{k-1}$ ∧ 
      $x_1, \ldots x_{k-2}, x_k$ ∈ $L_{k-1}$};
   3.2 for each transaction $t$ in D do
      begin //Check which k-itemsets are included in transaction $t$.
      let $Tem_t$ ← the $k$-itemsets in $t$ that are also contained in $Tem_k$;
      for each itemset $A$ in $Tem_t$ do
         let $A$.count ← $A$.count + 1;
      end
Procedure AllItemsOfInterest

(3.3) let $L_k \leftarrow \{c | c \in Tem_k \land (\text{supp}(c) = (c\text{.count}/|D|) \geq ms)\}$;
     let $N_k \leftarrow Tem_k - L_k$;

(3.4) // Prune all uninteresting $k$-itemsets in $L_k$
     for each itemset $i$ in $L_k$ do
       if NOT(fipi($I$)) then
         let $L_k \leftarrow L_k - \{I\}$;
       let $PL \leftarrow PL \cup L_k$;

(3.5) // Prune all uninteresting $k$-itemsets in $N_k$
     for each itemset $J$ in $N_k$ do
       if NOT(iipi($J$)) then
         let $N_k \leftarrow N_k - \{J\}$;
       let $NL \leftarrow NL \cup N_k$;
     end

(4) output $PL$ and $NL$;

(5) return.
Procedure AllItemsOfInterest

- E.g. run of the algorithm (ms=0.3, mi=0.05)

<table>
<thead>
<tr>
<th>TID</th>
<th>Items bought</th>
</tr>
</thead>
<tbody>
<tr>
<td>T1</td>
<td>{A, B, D}</td>
</tr>
<tr>
<td>T2</td>
<td>{A, B, C, D}</td>
</tr>
<tr>
<td>T3</td>
<td>{B, D}</td>
</tr>
<tr>
<td>T4</td>
<td>{B, C, D, E}</td>
</tr>
<tr>
<td>T5</td>
<td>{A, E}</td>
</tr>
<tr>
<td>T6</td>
<td>{B, D, F}</td>
</tr>
<tr>
<td>T7</td>
<td>{A, E, F}</td>
</tr>
<tr>
<td>T8</td>
<td>{C, F}</td>
</tr>
<tr>
<td>T9</td>
<td>{B, C, F}</td>
</tr>
<tr>
<td>T10</td>
<td>{A, B, C, D, F}</td>
</tr>
</tbody>
</table>

Table II. Single Frequent Items in $TD$

<table>
<thead>
<tr>
<th>Item</th>
<th>Number of Transactions</th>
<th>Support</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>5</td>
<td>0.5</td>
</tr>
<tr>
<td>B</td>
<td>7</td>
<td>0.7</td>
</tr>
<tr>
<td>C</td>
<td>5</td>
<td>0.5</td>
</tr>
<tr>
<td>D</td>
<td>6</td>
<td>0.6</td>
</tr>
<tr>
<td>E</td>
<td>3</td>
<td>0.3</td>
</tr>
<tr>
<td>F</td>
<td>5</td>
<td>0.5</td>
</tr>
</tbody>
</table>
Procedure AllItemsOfInterest

- Generate frequent and infrequent 2-itemset of interest.
  - When $ms = 0.3$, $L_2 = \{AB, AD, BC, BD, BF, CD, CF\}$, $N_2 = \{AC, AE, AF, BE, CE, DE, DF, EF\}$
- Use interest measure to prune.
Procedure AllItemsOfInterest

\[
f(A, B, ms, mi) = \frac{0.3 + 0.05 - (0.3 + 0.05) + 1}{|0.3 - 0.3| + |0.05 - 0.05| + 1} = 1
\]

\[
f(A, D, ms, mi) = \frac{0.3 + 0 - (0.3 + 0.05) + 1}{|0.3 - 0.3| + |0 - 0.05| + 1} < 1
\]

\[
f(B, C, ms, mi) = \frac{0.4 + 0.05 - (0.3 + 0.05) + 1}{|0.4 - 0.3| + |0.05 - 0.05| + 1} = 1
\]

\[
f(B, D, ms, mi) = \frac{0.6 + 0.18 - (0.3 + 0.05) + 1}{|0.6 - 0.3| + |0.18 - 0.05| + 1} = 1
\]

\[
f(B, F, ms, mi) = \frac{0.3 + 0.05 - (0.3 + 0.05) + 1}{|0.3 - 0.3| + |0.05 - 0.05| + 1} = 1
\]

\[
f(C, D, ms, mi) = \frac{0.3 + 0 - (0.3 + 0.05) + 1}{|0.3 - 0.3| + |0 - 0.05| + 1} < 1
\]

\[
f(C, F, ms, mi) = \frac{0.3 + 0.05 - (0.3 + 0.05) + 1}{|0.3 - 0.3| + |0.05 - 0.05| + 1} = 1
\]

- So AD and CD are not of interest, they are removed from L2.
Procedure AllItemsOfInterest

- So the resulting frequent 2-itemsets are as follows:

<table>
<thead>
<tr>
<th>Item</th>
<th>Number of Transactions</th>
<th>Support</th>
</tr>
</thead>
<tbody>
<tr>
<td>AB</td>
<td>3</td>
<td>0.3</td>
</tr>
<tr>
<td>BC</td>
<td>4</td>
<td>0.4</td>
</tr>
<tr>
<td>BD</td>
<td>6</td>
<td>0.6</td>
</tr>
<tr>
<td>BF</td>
<td>3</td>
<td>0.3</td>
</tr>
<tr>
<td>CF</td>
<td>3</td>
<td>0.3</td>
</tr>
</tbody>
</table>
Procedure AllItemsOfInterest

- Generate infrequent 2-itemsets using the iipi measure.
- Very similar to frequent 2-itemsets.

Table IV. Infrequent 2-Itemsets of Interest in NL

<table>
<thead>
<tr>
<th>Item</th>
<th>Number of Transactions</th>
<th>Support</th>
<th>Item</th>
<th>Number of Transactions</th>
<th>Support</th>
</tr>
</thead>
<tbody>
<tr>
<td>AC</td>
<td>2</td>
<td>0.2</td>
<td>AE</td>
<td>2</td>
<td>0.2</td>
</tr>
<tr>
<td>AF</td>
<td>2</td>
<td>0.2</td>
<td>BE</td>
<td>1</td>
<td>0.1</td>
</tr>
<tr>
<td>CE</td>
<td>1</td>
<td>0.1</td>
<td>DE</td>
<td>1</td>
<td>0.1</td>
</tr>
<tr>
<td>DF</td>
<td>2</td>
<td>0.2</td>
<td>EF</td>
<td>1</td>
<td>0.1</td>
</tr>
</tbody>
</table>
Extracting Positive and Negative Rules

- Continue like this to get all the itemsets.

<table>
<thead>
<tr>
<th>TID</th>
<th>Items bought</th>
<th>Algorithm iteration</th>
</tr>
</thead>
<tbody>
<tr>
<td>T1</td>
<td>{A,B,D}</td>
<td>Frequent 1-itemset A,B,C,D,E,F</td>
</tr>
<tr>
<td>T2</td>
<td>{A,B,C,D}</td>
<td>Frequent 2-itemset AB,BC,BD,BF,CF</td>
</tr>
<tr>
<td>T3</td>
<td>{B,D}</td>
<td>Infrequent 2-itemset AC,AE,AF,BE,CE,CF,DE,EF</td>
</tr>
<tr>
<td>T4</td>
<td>{B,C,D,E}</td>
<td></td>
</tr>
<tr>
<td>T5</td>
<td>{A,E}</td>
<td>Frequent 3-itemset BCD</td>
</tr>
<tr>
<td>T6</td>
<td>{B,D,F}</td>
<td>Infrequent 3-itemset BCF,BDF</td>
</tr>
<tr>
<td>T7</td>
<td>{A,E,F}</td>
<td></td>
</tr>
<tr>
<td>T8</td>
<td>{C,F}</td>
<td></td>
</tr>
<tr>
<td>T9</td>
<td>{B,C,F}</td>
<td></td>
</tr>
<tr>
<td>T10</td>
<td>{A,B,C,D,F}</td>
<td></td>
</tr>
</tbody>
</table>

Xindong Wu, University of Vermont, USA
Extracting Positive and Negative Rules

- Pruning strategy for rule generation: Piatetsky-Shapiro’s argument.

\[ \text{Dependence}(X, Y) = \frac{p(X \cup Y)}{p(X)p(Y)} = \frac{p(Y | X)}{p(Y)}. \]

- If Dependence(X,Y) = 1, X and Y are independent.
- If Dependence(X,Y) > 1, Y is positively dependent on X.
- If Dependence(X,Y) < 1, Y is negatively dependent on X (\sim Y is positively dependent on X).
Extracting Both Types of Rules

- Conditional probability increment ratio.

\[
CPIR(Y | X) = \begin{cases} 
\frac{p(Y|X) - p(Y)}{1-p(Y)}, & \text{if } p(Y|X) \geq p(Y), p(Y) \neq 1 \\
\frac{p(Y|X) - p(Y)}{p(Y)}, & \text{if } p(Y) > p(Y|X), p(Y) \neq 0
\end{cases}
\]

- Used to measure the correlation between X and Y.
  - When CPIR(X|Y)=0, X and Y are dependent.
  - When it is 1, they are perfectly correlated.
  - When it is -1, they are perfectly negatively correlated.
Extracting Both Types of Rules

Because $p(\neg A) = 1 - p(A)$, we only need the first half of the previous equation.

$$CPIR(Y|X) = \frac{P(Y|X) - P(Y)}{1 - P(Y)}$$

or

$$CPIR(Y|X) = \frac{\text{supp}(X \cup Y) - \text{supp}(X)\text{supp}(Y)}{\text{supp}(X)(1 - \text{supp}(Y))}$$

This value is used as confidence value.
3 Types of Negative Rules

Definition 1 in the paper:

- \( A \Rightarrow \neg B \) iff \( \text{supp}(A) \geq ms, \text{supp}(B) \geq ms, \text{interest}(A, \neg B) \geq mi, \) and \( \text{CPIR}(\neg B|A) \geq mc \)

- \( \neg A \Rightarrow B \) iff \( \text{supp}(A) \geq ms, \text{supp}(B) \geq ms, \text{interest}(\neg A, B) \geq mi, \) and \( \text{CPIR}(B|\neg A) \geq mc \)

- \( \neg A \Rightarrow \neg B \) iff \( \text{supp}(A) \geq ms, \text{supp}(B) \geq ms, \text{interest}(\neg A, \neg B) \geq mi, \) and \( \text{CPIR}(\neg B|\neg A) \geq mc \)
Algorithm PositiveAndNegative Associations

- **Input:** D – a database; minsupp, minconf, mininterest
- **Output:** Association rules
- **Step1:** calls procedure AllItemsetsOfInterest to generate the sets PL and NL with frequent and infrequent itemsets of interest respectively, in the database D.
- **Step2:** generates positive association rules of interest for an expression $X \cup Y$ of $A$ in PL if $\text{fipis}(X, Y)$. Check $\text{CPIR}(Y|X)$ and $\text{CPIR}(X|Y)$.
- **Step3:** generates negative association rules of interest for an expression $X \cup Y$ of $A$ in NL if $\text{iipis}(X, Y)$. Check $\text{CPIR}(\neg Y|X)$, $\text{CPIR}(\neg X|Y)$, $\text{CPIR}(X|\neg Y)$, $\text{CPIR}(Y|\neg X)$, $\text{CPIR}(\neg Y|\neg X)$ and $\text{CPIR}(\neg X|\neg Y)$. 

ICML-2002

Xindong Wu, University of Vermont, USA

32
Extracting rules

- One snapshot of an iteration in the algorithm

Example 5. For itemset $B \cup E$ in $NL$, $\text{supp}(B) = 0.7$, $\text{supp}(E) = 0.3$, $\text{supp}(-E) = 0.7$, $\text{supp}(B \cup -E) = 0.6$, and

$$\text{CPIR}(-E|B) = \frac{\text{supp}(B \cup -E) - \text{supp}(B)\text{supp}(-E)}{\text{supp}(B)(1 - \text{supp}(-E))} = \frac{0.6 - 0.7 \times 0.7}{0.7 \times (1 - 0.7)} = 0.524$$

Also,

$$f(B, -E, ms, mc, mi) = \frac{0.6 + 0.524 + 0.11 - (0.3 + 0.5 + 0.05) + 1}{|0.6 - 0.3| + |0.524 - 0.5| + |0.11 - 0.05| + 1} = 1$$

$$g(B, -E, ms, mc, mi) = f(B, -E, ms, mc, mi) + \frac{\text{supp}(B) + \text{supp}(E) - 2ms + 1}{|\text{supp}(B) - ms| + |\text{supp}(E) - ms| + 1}$$

$$= 1 + \frac{0.7 + 0.3 - 2 \times 0.3 + 1}{|0.7 - 0.3| + |0.3 - 0.3| + 1} = 2$$

- The result $B=>\sim E$ is a valid rule.
Experimental Results (1)

- A comparison with Apriori like algorithm without pruning
Experimental Results (2)

- A comparison with no-pruning

![Graph showing the comparison between MBP and MNP with different number of transactions.](image)
Experimental Results

- Effectiveness of pruning

![Running Efficiency Graph](image)

![Number of pii and nii Graphs](image)
Related Work

- Negative relationships between frequent itemsets, but not how to find negative rules (Brin, Motwani and Silverstein 1997)

- Strong negative association mining using domain knowledge (Savasere, Ommiecinski and Navathe 1998)
Conclusions

- Negative rules are useful
- Pruning is essential to find frequent and infrequent itemsets.
- Pruning is important to find negative association rules.
- There could be more negative association rules if you have different conditions.
1. List the three types of negative association rules. (see Definition 1)

- $A \Rightarrow \sim B$ iff $\text{supp}(A) \geq ms$, $\text{supp}(B) \geq ms$, $\text{interest}(A, \sim B) \geq mi$, and $\text{CPIR}(\sim B|A) \geq mc$

- $\sim A \Rightarrow B$ iff $\text{supp}(A) \geq ms$, $\text{supp}(B) \geq ms$, $\text{interest}(\sim A, B) \geq mi$, and $\text{CPIR}(B|\sim A) \geq mc$

- $\sim A \Rightarrow \sim B$ iff $\text{supp}(A) \geq ms$, $\text{supp}(B) \geq ms$, $\text{interest}(\sim A, \sim B) \geq mi$, and $\text{CPIR}(\sim B|\sim A) \geq mc$

Or use definition in the paper.
Exam questions

2. Why are infrequent itemsets necessary for negative association mining?

Negative association rules of form \( A \Rightarrow \neg B \) means \( \text{supp}(A \cup \neg B) \geq ms. \) \( \text{supp}(A \cup B) = \text{supp}(A) - \text{supp}(A \cup \neg B). \) For most cases, the \( \text{supp}(A) < 2*ms. \) Therefore \( \text{supp}(A \cup B) < ms, \) which means \( A \cup B \) is infrequent itemsets. So to find negative association rules, we need to find infrequent itemsets first.
3. When does pruning take place and what measurements can be used?

- Pruning happens in itemsets generation process and rule extraction process.
- There are three measures for pruning.
  - The first is interest($X, Y$). It is used for itemsets generation.
  - The second measure is $\text{supp}(X) > \text{ms}$, $\text{supp}(Y) > \text{ms}$. It is used for infrequent itemsets.
  - The third is $\text{CPIR}(Y|X)$. It is used for rule extraction.