PARMA: A Parallel Randomized Algorithm for Approximate Association Rules Mining in MapReduce

Date: 2013/10/16
Source: CIKM’12
Outline

• Introduction
• Approach
• Experiment
• Conclusion
Association Rules

• Market-Basket transactions

<table>
<thead>
<tr>
<th>TID</th>
<th>Items</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Bread, Milk</td>
</tr>
<tr>
<td>2</td>
<td>Bread, Diaper, Beer, Eggs</td>
</tr>
<tr>
<td>3</td>
<td>Milk, Diaper, Beer, Coke</td>
</tr>
<tr>
<td>4</td>
<td>Bread, Milk, Diaper, Beer</td>
</tr>
<tr>
<td>5</td>
<td>Bread, Milk, Diaper, Coke</td>
</tr>
</tbody>
</table>

Example of Association Rules

\{\text{Diaper}\} \rightarrow \{\text{Beer}\},
\{\text{Milk, Bread}\} \rightarrow \{\text{Eggs, Coke}\},
\{\text{Beer, Bread}\} \rightarrow \{\text{Milk}\},
Itemset

- \( I = \{\text{Bread, Milk, Diaper, Beer, Eggs, Coke}\} \)
- Itemsets
  - 1-itemsets: \{Beer\}, \{Milk\}, \{Bread\}, ...
  - 2-itemsets: \{Bread, Milk\}, \{Bread, Beer\}, ...
  - 3-itemsets: \{Milk, Eggs, Coke\}, \{Bread, Milk, Diaper\}, ...

- \( t_1 \) contains \{Bread, Milk\}, but doesn’t contain \{Bread, Beer\}
Frequent Itemset

• **Support count**: $\sigma(X)$
  - Frequency of occurrence of an itemset $X$
  - $\sigma(X) = |\{t_i \mid X \subseteq t_i, t_i \in T\}|$
  - E.g. $\sigma(\{\text{Milk, Bread, Diaper}\}) = 2$

• **Support**
  - Fraction of transactions that contain an itemset $X$
  - $s(X) = \sigma(X) / |T|$
  - E.g. $s(\{\text{Milk, Bread, Diaper}\}) = 2/5$

• **Frequent Itemset**
  - An itemset $X$ $s(X) \geq \text{minsup}$
Association Rule

- **Association Rule**
  - X → Y, where X and Y are itemsets
  - Example: 
    {Milk, Diaper} → {Beer}

- **Rule Evaluation Metrics**
  - **Support**
    - Fraction of transactions that contain both X and Y
    - \( s(X \rightarrow Y) = \frac{\sigma(X \cup Y)}{|T|} \)
  - **Confidence**
    - How often items in Y appear in the transactions that contain X
    - \( c(X \rightarrow Y) = \frac{\sigma(X \cup Y)}{\sigma(X)} \)

<table>
<thead>
<tr>
<th>TID</th>
<th>Items</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Bread, Milk</td>
</tr>
<tr>
<td>2</td>
<td>Bread, Diaper, Beer, Eggs</td>
</tr>
<tr>
<td>3</td>
<td>Milk, Diaper, Beer, Coke</td>
</tr>
<tr>
<td>4</td>
<td>Bread, Milk, Diaper, Beer</td>
</tr>
<tr>
<td>5</td>
<td>Bread, Milk, Diaper, Coke</td>
</tr>
</tbody>
</table>

Example:

\( \{\text{Milk, Diaper}\} \rightarrow \{\text{Beer}\} \)

\[ s = \frac{\sigma(\text{Milk, Diaper, Beer})}{|T|} = \frac{2}{5} = 0.4 \]
\[ c = \frac{\sigma(\text{Milk, Diaper, Beer})}{\sigma(\text{Milk, Diaper})} = \frac{2}{3} = 0.67 \]
Association Rule Mining Task

- Given a set of transactions $T$, the goal of association rule mining is to find all rules having
  - $\text{support} \geq \text{minsup}$
  - $\text{confidence} \geq \text{minconf}$
Goal

• Because:
  • Number of transactions
  • Cost of the existing algorithm, e.g. Apriori, FP-Tree
  • What can we do in big data?
    • Sampling
    • Parallel

• Goal:
  • A MapReduce algorithm for discovering approximate collections of frequent itemsets or association rules
Outline

• Introduction
• Approach
• Experiment
• Conclusion
Sampling

Original Data → Sampling → Find FI in Sample

Question: Is the sample always good?
Definition

\[ \text{Fl}(D, I, \theta) = \{(A, f_D(A)) : A \in 2^I \text{ and } f_D(A) \geq \theta\}. \]

\[ \text{TOPK}(D, I, K) = \text{Fl}(D, I, f_D^{(K)}). \]  \hspace{1cm} (1)

\((\varepsilon_1, \varepsilon_2)\) approximation of \(\text{Fl}(D, I, \theta)\) is a set
\[ C = \{(A, f_A, K_A) : A \in 2^I, f_A \in K_A \subseteq [0,1]\} \]

1. \(C\) contains all itemsets appearing in \(\text{Fl}(D, I, \theta)\);
2. \(C\) contains no itemset \(A\) with frequency \(f_D(A) < \theta - \varepsilon_1\);
3. For every triplet \((A, f_A, K_A) \in C\), it holds
   (a) \(|f_D(A) - f_A| \leq \varepsilon_2\).
   (b) \(f_A\) and \(f_D(A)\) belong to \(K_A\).
   (c) \(|K_A| \leq 2\varepsilon_2\).
How many samples do we need?

**Lemma 1.** [29, Lemma 1] Let \( \mathcal{D} \) be a dataset with transactions built on an alphabet \( \mathcal{I} \), and let \( d \) be the maximum integer such that \( \mathcal{D} \) contains at least \( d \) transactions of size at least \( d \). Let \( 0 < \varepsilon, \delta, \theta < 1 \). Let \( S \) be a random sample of \( \mathcal{D} \) containing \( |S| = \frac{2}{\varepsilon^2} \left( d + \log \frac{1}{\delta} \right) \) transactions drawn uniformly and independently at random with replacement from those in \( \mathcal{D} \), then with probability at least \( 1 - \delta \), the set \( \text{Fl}(S, \mathcal{I}, \theta - \frac{\varepsilon}{2}) \) is a \( (\varepsilon, \varepsilon/2) \)-approximation of \( \text{Fl}(\mathcal{D}, \mathcal{I}, \theta) \).
Introduction of MapReduce
Concept

Total Data

Sampling locally
Sampling locally
Sampling locally
Sampling locally
Sampling locally
Sampling locally

Mining locally
Mining locally
Mining locally
Mining locally
Mining locally
Mining locally

Global Frequent Itemset
Figure 1: A system overview of PARMA. Ellipses represent data, squares represent computations on that data and arrows show the movement of data through the system.
Parameter Space

- $p$: number of processors/nodes
- $m$: memory within each node
- $w$: sample size
- $N$: number of samples
- $\varepsilon$: error probability
- $\delta$: confidence bound

Given a fixed $\varepsilon$ and $\delta$ value we can measure the sample size using Lemma 1. If the sample size is greater than $m$ we have to increase the number of samples.
Trade-offs

- **Variables:** non-negative integer $N$, real $\phi \in (0, 1)$,
- **Objective:** minimize $2N/\varepsilon^2(d + \log(1/\phi))$.

\[
N \leq p \\
\phi \geq e^{-m\varepsilon^2/2+d} \\
N(1 - \phi) - \sqrt{N(1 - \phi)2\log(1/\delta)} \geq N/2 + 1
\]
In Reduce 2

• For each itemset, we have

\[ \mathcal{F}_A = (f_{S_i}(A), [f_{S_i}(A) - \varepsilon / 2, f_{S_i}(A) + \varepsilon / 2]) \]

• Then we use

\[ R = N(1 - \phi) - \sqrt{N(1 - \phi)2 \log(1/\delta)}. \tag{5} \]
Result

• The itemset $A$ is declared globally frequent and will be present in the output if and only if $|\mathcal{F}_A| \geq R$

• Let $[a_A, b_A]$ be the shortest interval such that there are at least $N-R+1$ elements from $\mathcal{F}_A$ that belong to this interval.

$$\tilde{f}(A) = a_A + \frac{b_A - a_A}{2}$$

$$\mathcal{K}_A = \left[a_A - \frac{\varepsilon}{2}, b_A + \frac{\varepsilon}{2}\right]$$

$$(A, (\tilde{f}(A), \mathcal{K}_A))$$
Lemma 2. [29, Lemma 6] Let \( \mathcal{D} \) be a dataset with transactions built on an alphabet \( \mathcal{I} \), and let \( d \) be the maximum integer such that \( \mathcal{D} \) contains at least \( d \) transactions of size at least \( d \). Let \( 0 < \varepsilon, \delta, \theta, \gamma < 1 \) and let \( \varepsilon_{rel} = \frac{\varepsilon}{\max\{\theta, \gamma\}} \). Fix \( c > 4 - 2\varepsilon_{rel} \), \( \eta = \frac{\varepsilon_{rel}}{c} \), and \( p = \frac{1 - \eta}{1 + \eta} \theta \). Let \( S \) be a random sample of \( \mathcal{D} \) containing \( \frac{1}{\eta^2 p} (d \log \frac{1}{p} + \log \frac{1}{\delta}) \) transactions from \( \mathcal{D} \) sampled independently and uniformly at random. Then \( \text{AR}(S, \mathcal{I}, (1 - \eta)\theta, \frac{1 - \eta}{1 + \eta} \gamma) \) is an \((\varepsilon, \varepsilon/2)\) approximation to \( \text{AR}(\mathcal{D}, \mathcal{I}, \theta, \gamma) \).
Outline

- Introduction
- Approach
- Experiment
- Conclusion
Implementation

- Amazon Web Service: ml.xlarge - 17GB
- Hadoop with 8 nodes
- Parameters:
  \[ \varepsilon = 0.05 \text{ and } \delta = 0.01 \]
- Compare against DistCount, PFP

<table>
<thead>
<tr>
<th>number of items</th>
<th>1000</th>
</tr>
</thead>
<tbody>
<tr>
<td>average length</td>
<td>5</td>
</tr>
<tr>
<td>average size of maximal potentially large itemsets</td>
<td>5</td>
</tr>
<tr>
<td>number of maximal potentially large itemsets</td>
<td>5</td>
</tr>
<tr>
<td>correlation among maximal potentially large itemsets</td>
<td>0.1</td>
</tr>
<tr>
<td>corruption of maximal potentially large itemsets</td>
<td>0.1</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>number of items</th>
<th>10000</th>
</tr>
</thead>
<tbody>
<tr>
<td>average length</td>
<td>10</td>
</tr>
<tr>
<td>average size of maximal potentially large itemsets</td>
<td>5</td>
</tr>
<tr>
<td>number of maximal potentially large itemsets</td>
<td>20</td>
</tr>
<tr>
<td>correlation among maximal potentially large itemsets</td>
<td>0.1</td>
</tr>
<tr>
<td>corruption of maximal potentially large itemsets</td>
<td>0.1</td>
</tr>
</tbody>
</table>
Compare with other Algorithm

Figure 2: A runtime comparison of PARMA with DistCount (top) and PFP (bottom).
Figure 3: A comparison of runtimes of the map/reduce/shuffle phases of PARMA, as a function of number of transactions. Run on an 8 node Elastic MapReduce cluster.
Acceptable False Positives

<table>
<thead>
<tr>
<th>$\theta$</th>
<th>Real FI’s</th>
<th>Output AFP’s</th>
<th>Max AFP’s</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.06</td>
<td>11016</td>
<td>11797</td>
<td>201636</td>
</tr>
<tr>
<td>0.09</td>
<td>2116</td>
<td>4216</td>
<td>10723</td>
</tr>
<tr>
<td>0.12</td>
<td>1367</td>
<td>335</td>
<td>1452</td>
</tr>
<tr>
<td>0.15</td>
<td>1053</td>
<td>299</td>
<td>415</td>
</tr>
</tbody>
</table>

Table 3: **Acceptable False Positives** in the output of PARMA
Error in frequency estimations

Figure 7: Error in frequency estimations as frequency varies.

Figure 8: Width of the confidence intervals as frequency varies.
Outline

• Introduction
• Approach
• Experiment
• Conclusion
Conclusion

• A parallel algorithm for mining quasi-optimal collections of frequent itemsets and association rules in MapReduce.
• 30-55% runtime improvement over PFP.
• Verify the accuracy of the theoretical bounds, as well as show that in practice our results are orders of magnitude more accurate than is analytically guaranteed.