Thus, communication cost is comparable to I/O cost of a-priori. Loosely – For every sub-itemset, hidden data values sent only possible solutions infinite!

Security is symmetric

The overall algorithm

1. \( L_1 = \{ \text{large 1-against} \} \)
2. for \( (k=2; L_{k-1} \neq \emptyset; k++) \)
3. \( C_k = \text{apriori-gen}(L_{k-1}) \)
4. for all candidates \( c \in C_k \) do begin
5. if all the attributes in \( c \) are exactly at A or B
6. that party independently calculates \( c \) count
7. else
8. let A have \( l \) of the attributes and B have the remaining \( m \) attributes
9. construct \( X \) on A's side and \( Y \) on B's side where \( X = \prod_{i=1}^{l} \tilde{A}_i \) and \( Y = \prod_{i=1}^{m} \tilde{B}_i \)
10. compute \( c \) count = \( X \cdot Y = \sum_{i=1}^{n} a_i + y_i \)
11. end if
12. \( L_k = L_k \cup c \cdot c \) count \( \geq \) minsup
13. end if
14. end for
15. Answer = \( \cup L_k \)

Security and Communication Analysis

- For Component Protocol,
  - A sends to B \( n+1 \) equations in \( 3n/2 \) unknowns
  - B sends to A \( n \geq 1 \) equations in \( n \) unknowns
- Total communication = \( 3n/2 \) values (3 messages)

- Security is symmetric
- Based on revealing less equations than the number of unknowns – possible solutions infinite!
- Everything revealed only when about half the values are (externally) revealed
- For every sub-itemset, hidden data values sent only once.

Thus, communication cost is comparable to I/O cost of a-priori. Loosely – Communication Cost = \( O(\text{a-priori I/O cost}) \), \( 1.5c < k < 2.5 \)

The data model

- Two parties – Alice (A) and Bob (B)
- Same set of entities (data cleansing, join assumed done)
- A has \( p \) attributes, \( A_1 \ldots A_p \)
- B has \( q \) attributes, \( B_1 \ldots B_q \)
- Total number of transactions, \( n \)
- Support Threshold, \( k \)

The Solution

The Component Protocol (Simplified Version)

- A generates \( n/2 \) randoms, \( R_1 \ldots R_{n/2} \)
- A sends the following \( n \) values to B
  \[
  \begin{align*}
    x_i \cdot a_1 + R_1 + R_2 + L + a_{3/2} \cdot R_3 \\cdots \\cdots \\
    \vdots \\
    x_i \cdot a_{n/2} + R_{n/2} + L + a_{3/2} \cdot R_3 \\
  \end{align*}
  \]
- B multiplies each value he gets with the corresponding y value he has and adds all of them up to get a sum \( S \) which he sends to A.
  \[
  S = \sum x_i \cdot y_i + R_1 + R_2 + \cdots \cdots \cdots + R_{n/2} \]

- Group the \( x_i \cdot y_i \) terms, and expand the equations

- A already knows \( R_1 \ldots R_{n/2} \)
- Now, if B sends these \( n/2 \) values to A, A can remove the baggage and get the scalar product

Limitations

- Protocol works only in Honest-But-Curious model
  - Either party can “doctor” their input and find out specific information
  - Can be fixed by allowing approximations on both sides (Thus neither party can get specific information, but the final result will be approximate)

- \( \{0,1\} \) data causes problems with security

Work Completed

- Extended to multiple parties
- Resistant to collusion
- Solutions for clustering and classification also developed

Prior/Related Work

- Privacy Preserving Data Mining
  - Data Perturbation, Secure Multiparty Computation Approaches
- Distributed Data Mining
  - Operating on both Horizontally and Vertically partitioned data
  - Meta learning approaches
- Secure Multiparty Computation
  - General proofs, Some specific problems solved