Privacy-Preserving Collaborative Data Mining

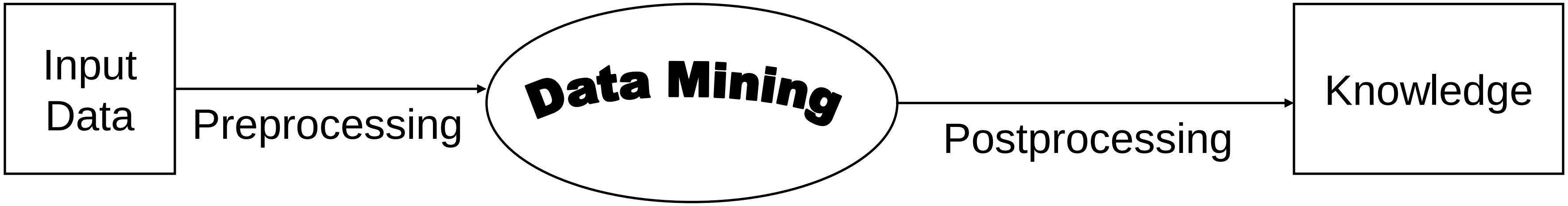
Justin Zhan
justinzh@andrew.cmu.edu or
justinzzhan@gmail.com

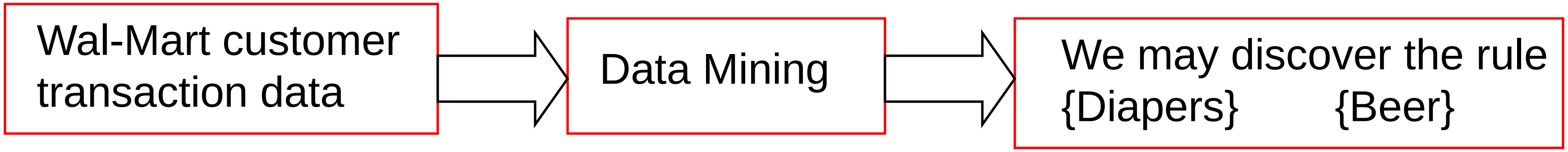
Overview

- Privacy Learning Library
- Efficient Privacy-Preserving Collaborative Compiler System Using Scalar Product
- Social Computing

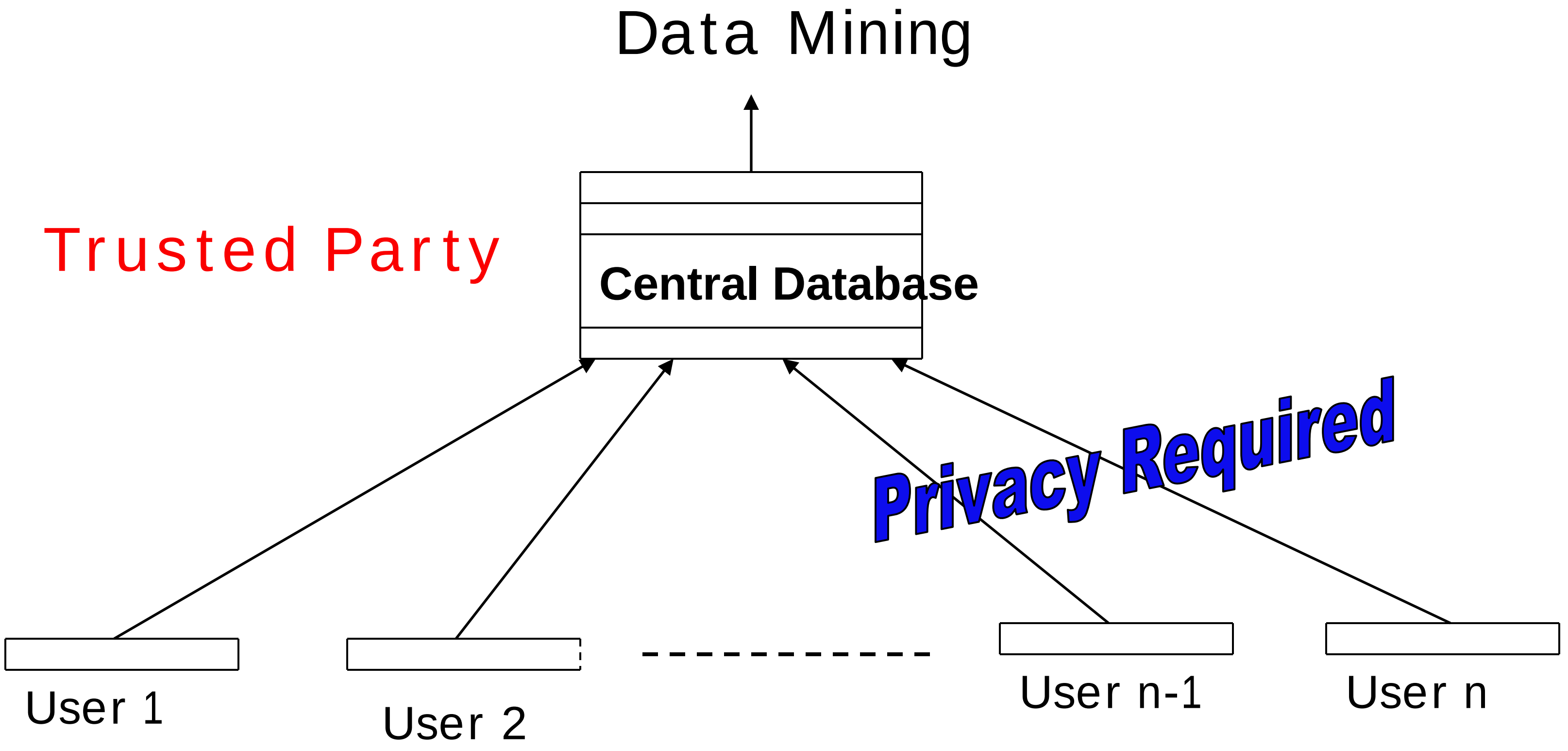
What Data Mining Is

- Data mining is the process of automatically discovering useful knowledge in large databases.

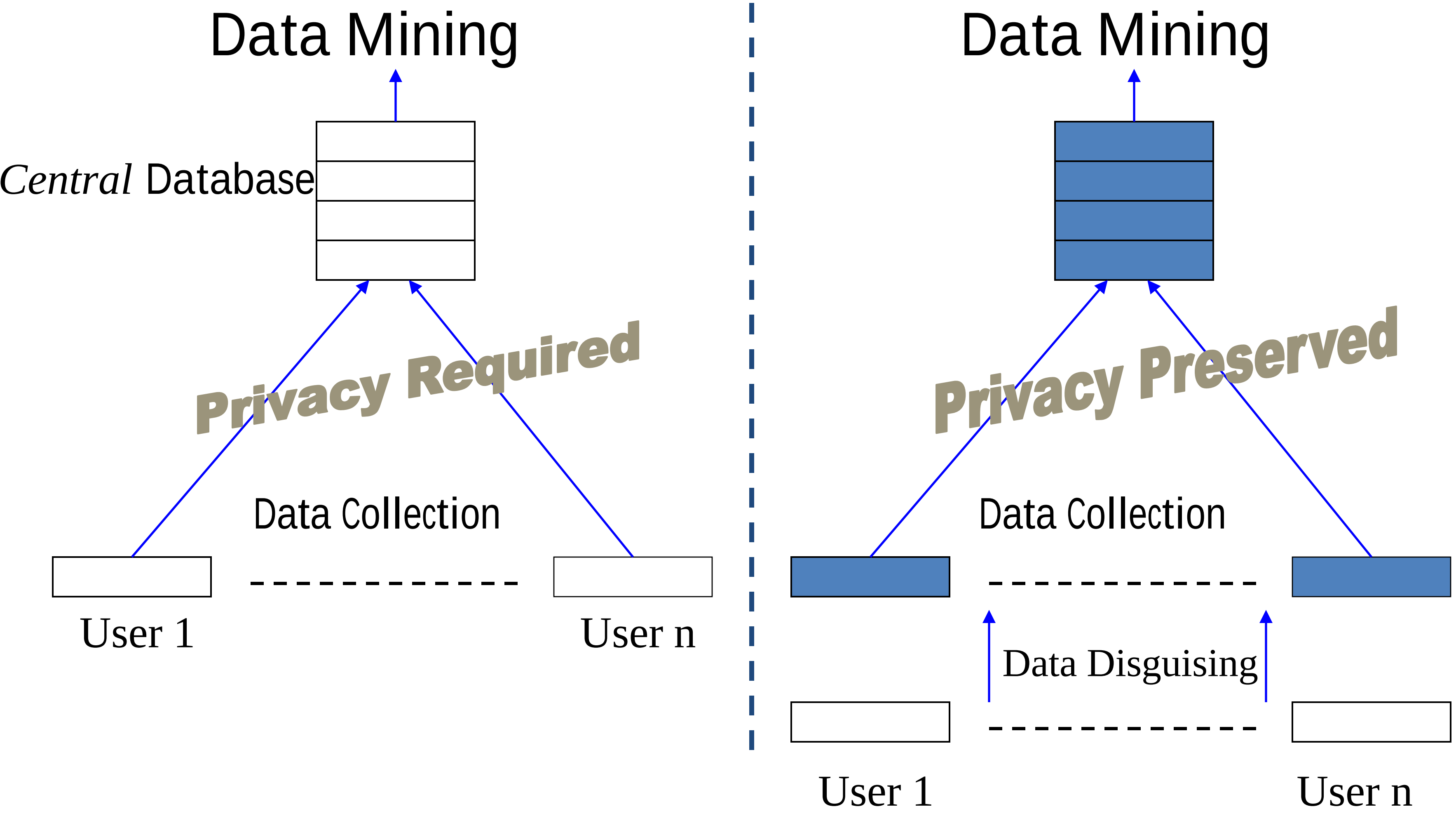




A Trusted-Party Model



Privacy Protection



Randomized Response Techniques

- An example:

  Survey: how many people have ever driven while intoxicated?

- People may not want to divulge their information
- How to conduct such a survey?
- Two related questions are asked for each person

  1. Is it true that you have ever driven while intoxicated?
  2. Is it true that you haven’t ever driven while intoxicated?

- Each person randomly selects one question to answer
  - Probability of selecting question 1 is .
  - Probability of selecting question 2 is (1 -).

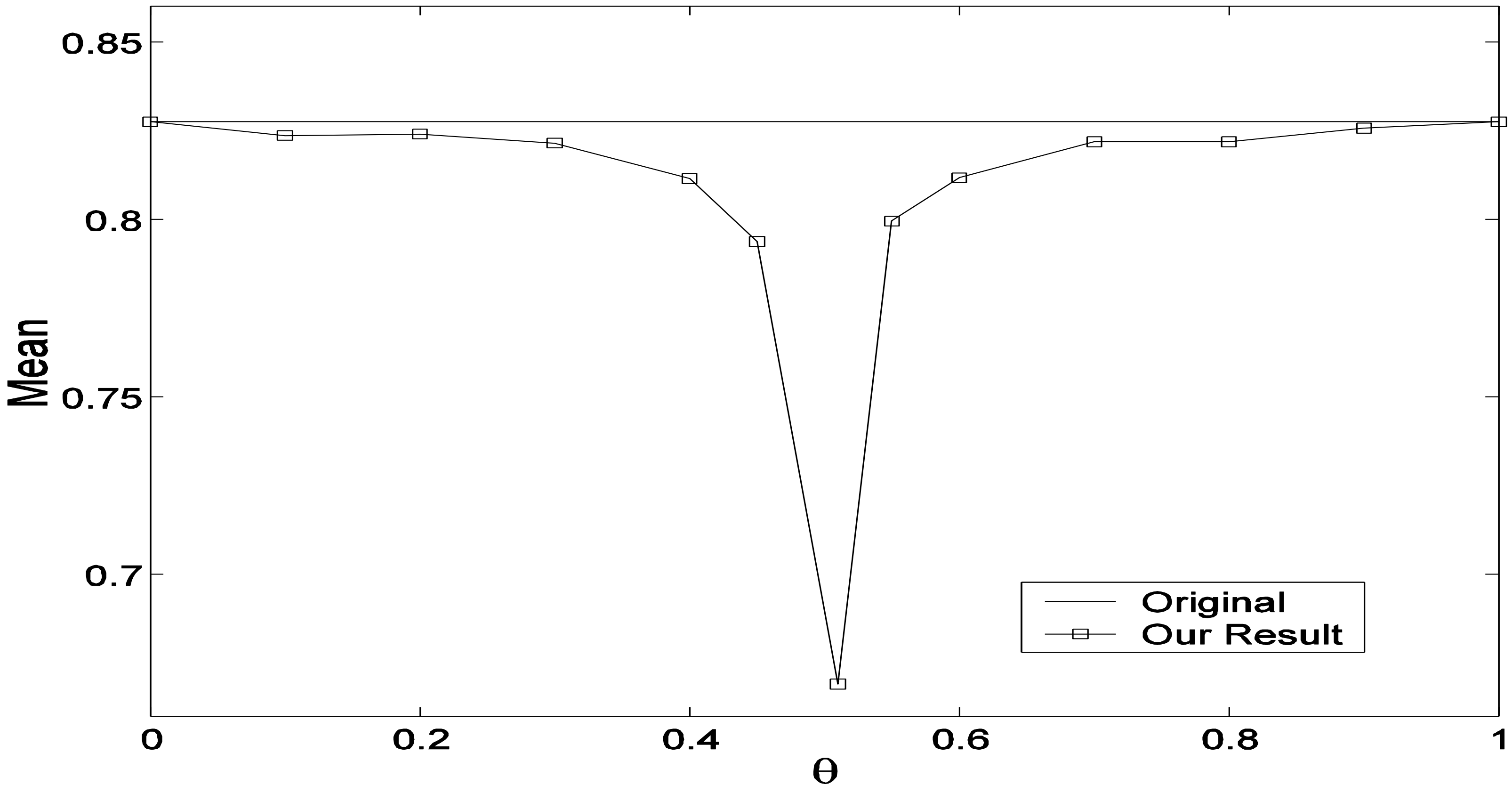
How Randomized Response Works

$$P^{*}(A = yes) = P(A = yes) \quad + P(A=no)(1- \quad) \quad (1)$$

$$P^{*}(A = no) = P(A = no) \quad + P(A=yes)(1- \quad) \quad (2)$$

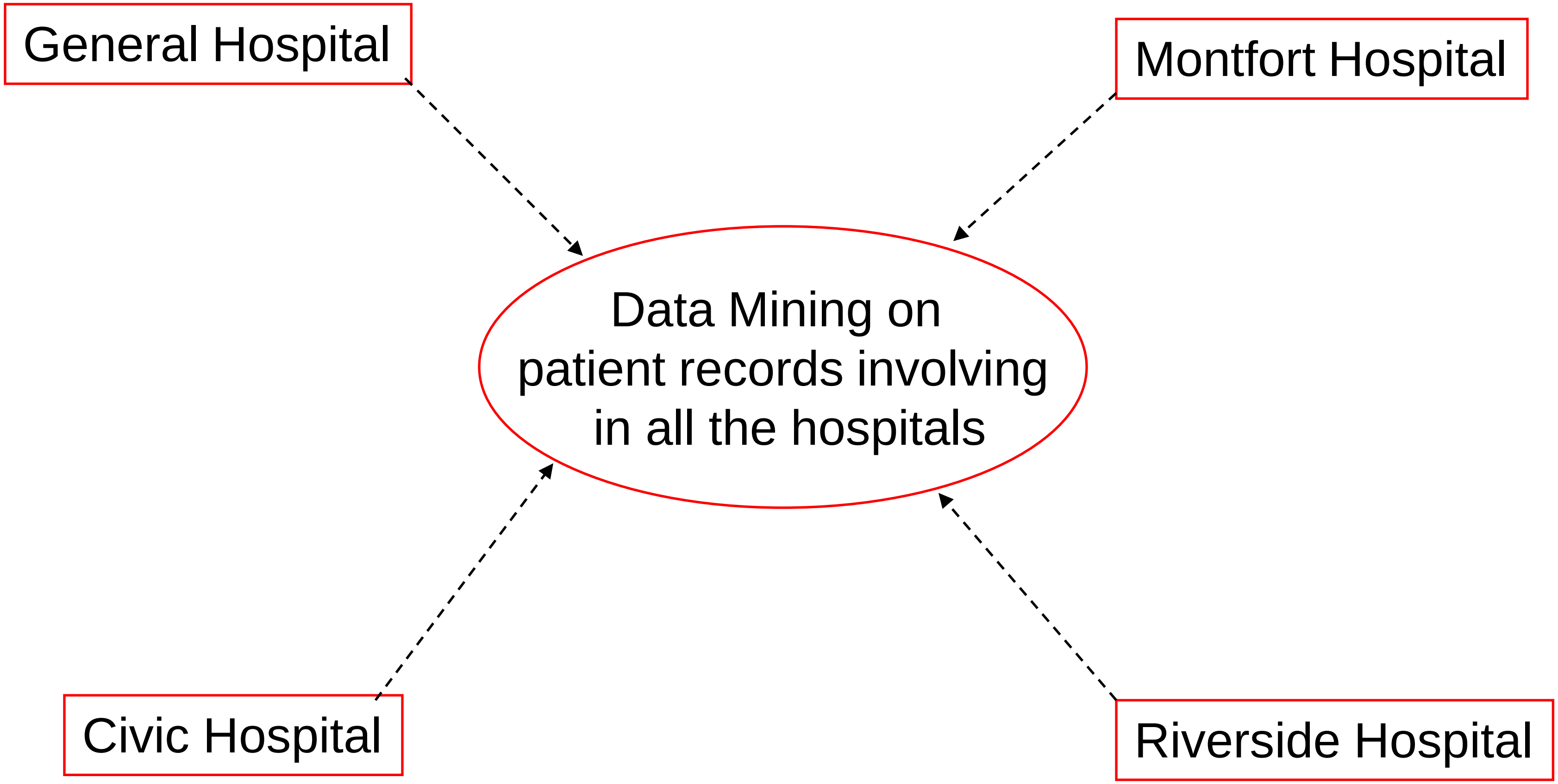
- $P^{*}(A = yes)$ and $P^{*}(A = no)$: directly count the disguised data.
- $P(A = yes)$: The percentage of people who have driven while intoxicated.
- Solving Eq. (1) and (2), we get $P(A = yes)$.

Experimental Results



Why PPDM is Important

Biomedical Computing



General Hospital

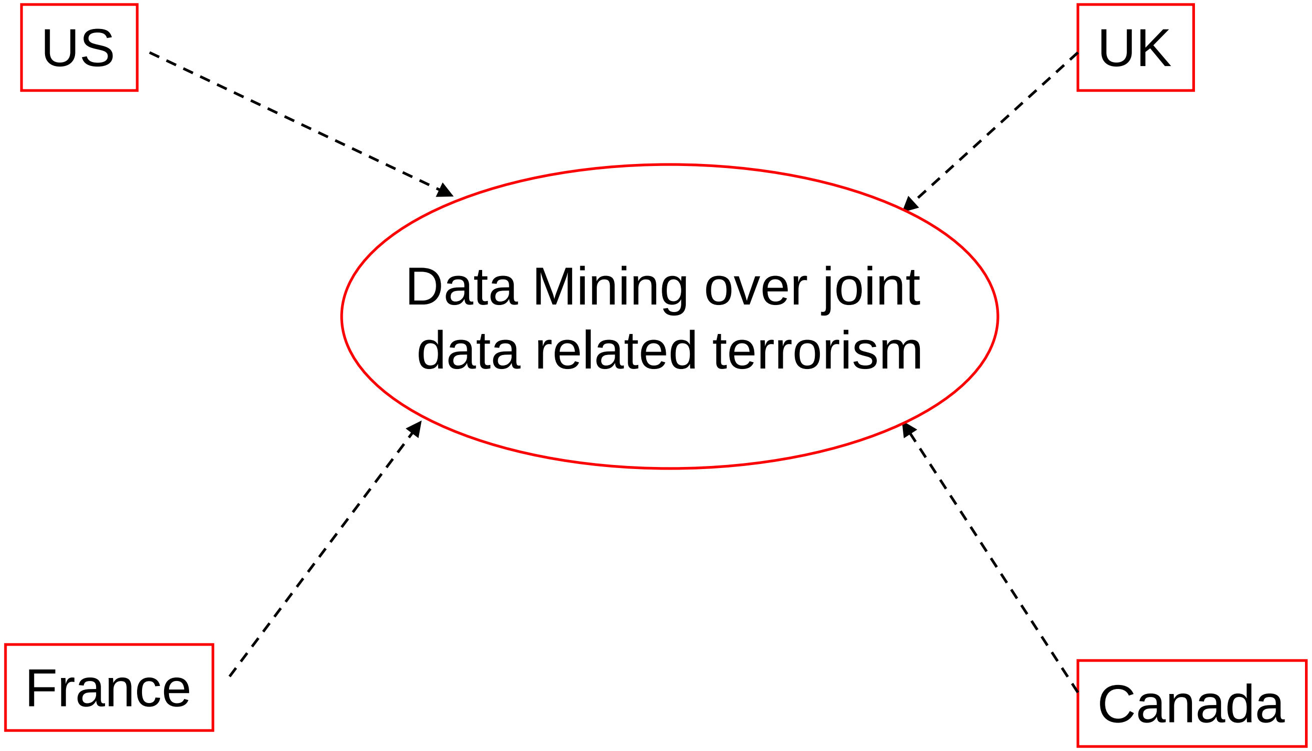
Montfort Hospital

Data Mining on patient records involving in all the hospitals

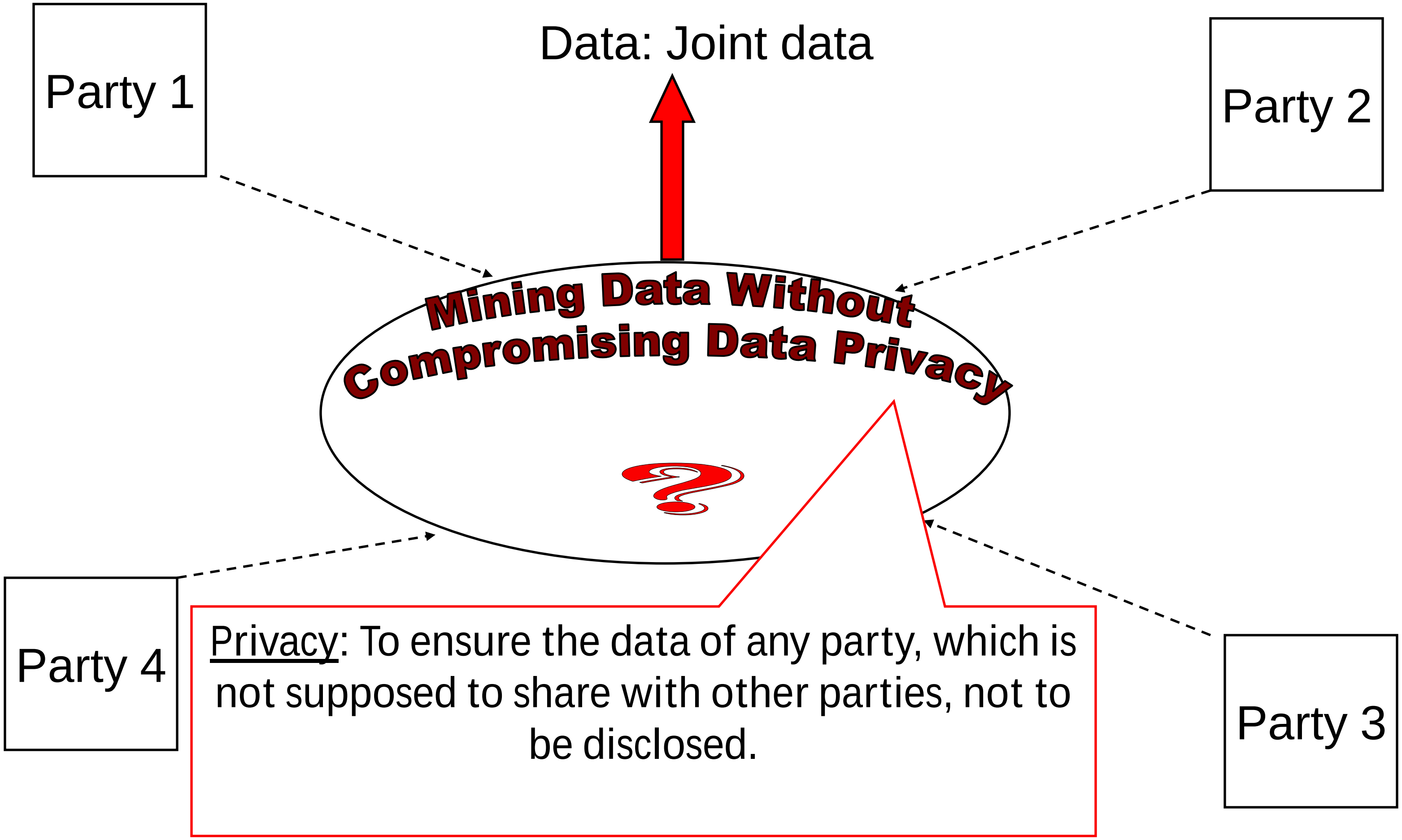
Civic Hospital

Riverside Hospital

Government Collaboration (NATO)



Privacy-Preserving Collaborative Data Mining



Party 1

Data: Joint data

Party 2

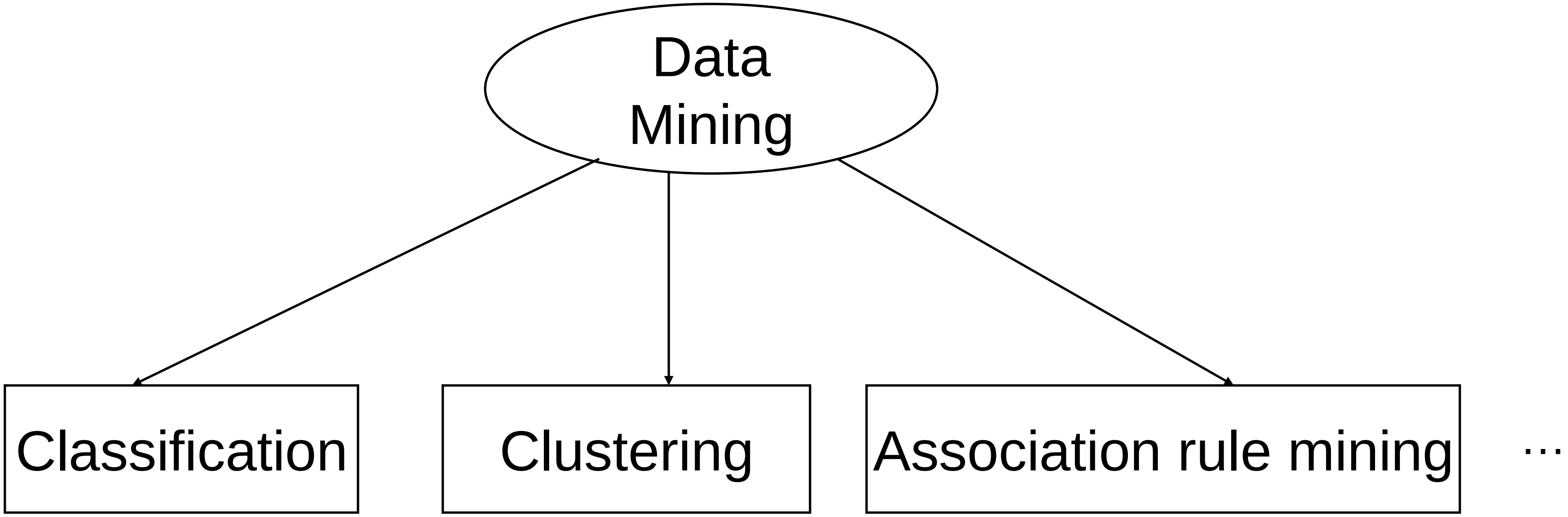
Mining Data Without Compromising Data Privacy

Party 4

Party 3

Privacy: To ensure the data of any party, which is not supposed to share with other parties, not to be disclosed.

Data Mining Tasks


Data Mining

Classification

Clustering

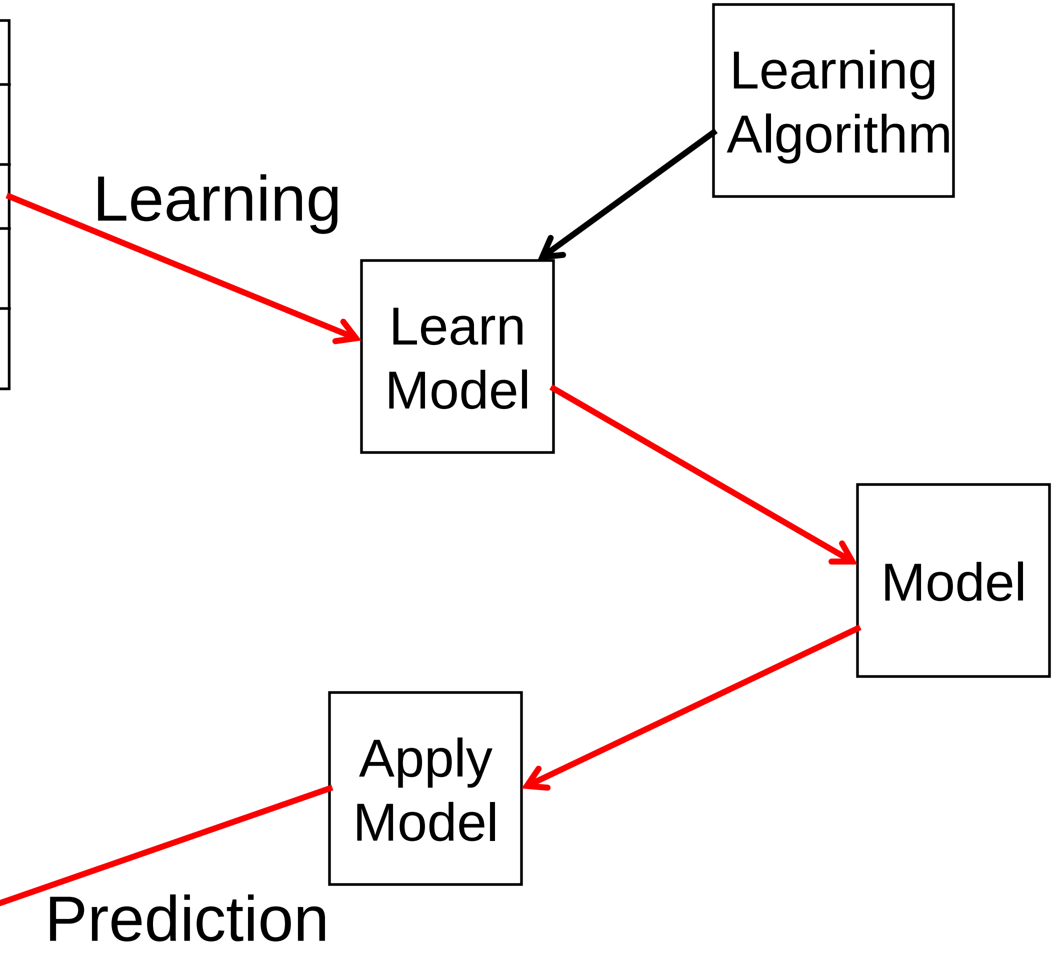
Association rule mining

…

Classification

Training Set

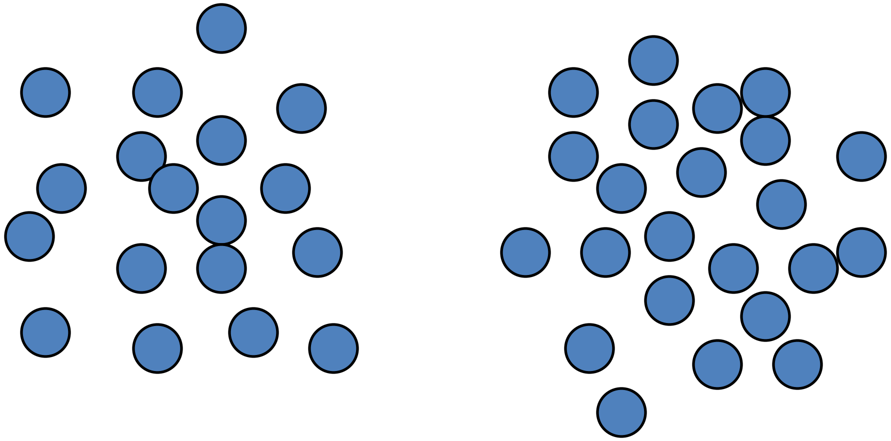
Tid	Attrib1	Attrib2	Class
1	Yes	Large	No
2	No	Medium	No
3	No	Large	Yes
4	No	Small	Yes



Test Set

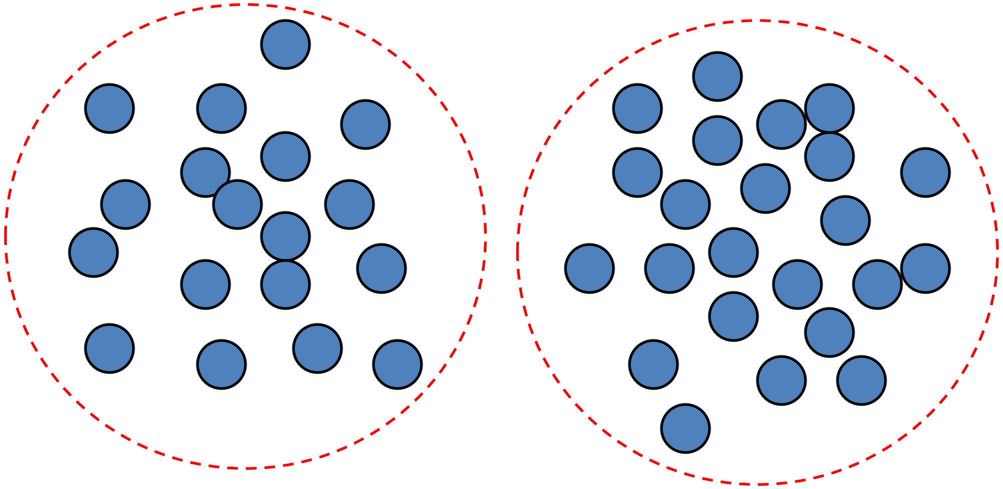
Tid	Attrib1	Attrib2	Class
1	No	Large	?
2	Yes	Large	?

Clustering



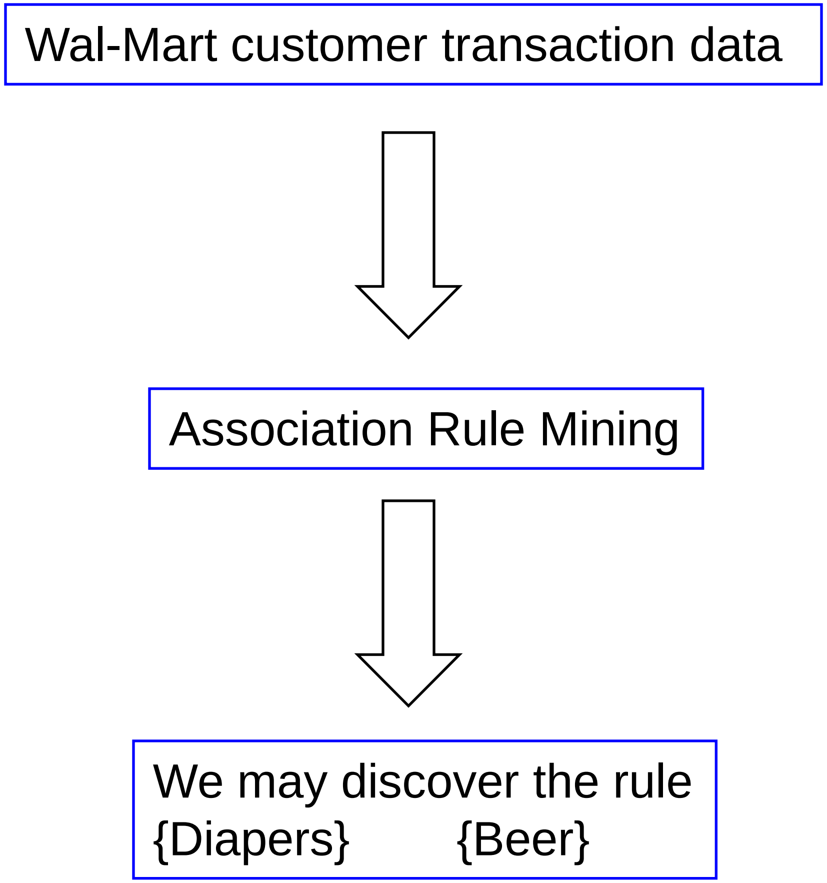
The data objects

Clustering



Two clusters

Association Rule Mining



Wal-Mart customer transaction data

Association Rule Mining

We may discover the rule
{Diapers} {Beer}

Association Rule Mining

TID	Items
1	{Bread, Milk}
2	{Bread, Diapers, Beer, Eggs}
3	{Milk, Diapers, Beer, Coke}
4	{Bread, Milk, Diapers, Beer}
5	{Bread, Milk, Diapers, Coke}

An example of market basket transactions.

TID: Transaction ID

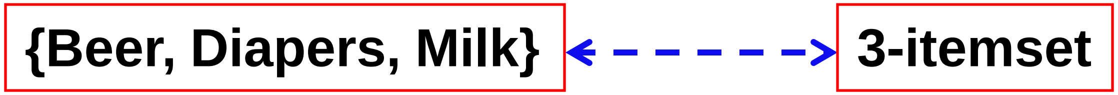
Binary Representation

The List of Attributes

TID	Bread	Milk	Diapers	Beer	Eggs	Coke
1	1	1	0	0	0	0
2	1	0	1	1	1	0
3	0	1	1	1	0	1
4	1	1	1	1	0	0
5	1	1	1	0	0	1

Itemset

- Let $I \quad \{i_1, i_2 \quad , i_d\}$ be the set of all items in a store and $T \quad \{t_1, t_2, \quad , t_N\}$ be the set of all transactions.
- Each transaction t_i contains a subset of items chosen from I
- In association rule mining, a collection of zero or more items is termed an itemset.
- If an itemset contains k items, it is called a k-itemset.

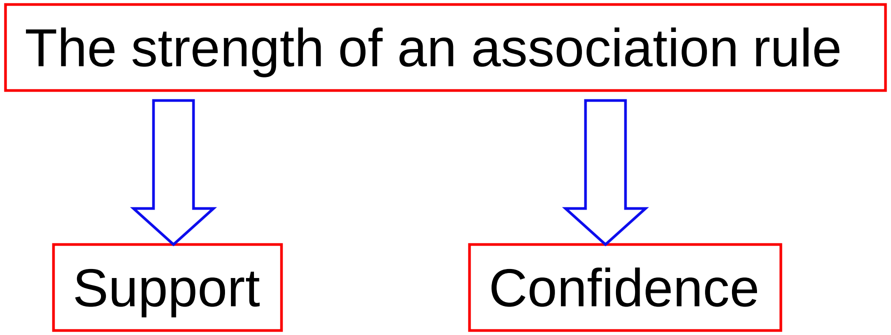


{Beer, Diapers, Milk} **3-itemset**

Association Rule

- An association rule is an expression of the form X Y, where X and Y are different sets of items.

{Bread} {Beer}



Support and Confidence

- Support determines how often a rule can be applied to a given data set.
- Confidence determines how frequently items in Y appear in transactions that contain X.

$$Support, \quad s(X \qquad Y) \quad \Pr(X \quad Y)$$

$$Confidence, \quad c(X \qquad Y) \quad \frac{\Pr(X \quad Y)}{\Pr(X)}$$

Association Rule Mining

Given a set of transactions T, find all the rules having support *minsup*, and confidence *minconf*, where *minsup* and *minconf* are the corresponding support and confidence thresholds.

Privacy-Preserving Collaborative Association Rule Mining

To enable multiple parties to conduct association rule mining over their joint data sets without disclosing their private data.

Horizontal Collaboration

	TID	Bread	Milk	Diapers	Beer	Eggs	Coke
Alice	1	1	1	0	0	0	0
	2	1	0	1	1	1	0
	3	0	1	1	1	0	1
Bob	4	1	1	1	1	0	0
	5	1	1	1	0	0	1

Vertical Collaboration

Store 1

TID	Bread	Milk	Diapers
1	1	1	0
2	1	0	1
3	0	1	1
4	1	1	1
5	1	1	1

Alice

This Talk Focuses on Vertical Collaboration

Store 2

TID	Beer	Wine
1	1	0
2	1	1
3	0	1
4	1	1
5	1	1

Bob

Association Rule Mining Algorithm

[Agrawal et al. 1993]

```
$L_1$ = large 1-itemsets
for $C_k \quad apriori \quad gen(L_{k \ 1})$ do begin
    $(k \quad 2; L_{k \ 1} \qquad ; k \qquad )$
        for all candidates $c \quad C_k$ do begin
            compute c.count
        end
    $L_k \quad \{c \quad C_k \mid c.count \quad \min \quad sup\}$
end
Return $L \qquad {}_k L_k$
```

c.count is the frequency count for a given itemset.

Key issue: to compute the frequency count, *we* needs to access attributes that belong to different parties.

Frequent Itemset Generation

Minimum support count = 3

Candidate 1-Itemsets

Item	Count
Beer	3
Bread	4
Diapers	4
Milk	4

Candidate Large 1-Itemsets

Item	Count
Beer	3
Bread	4
Diapers	4
Milk	4

REMOVED

Frequent Itemset Generation

Minimum support count = 3

Candidate
Large 1-Itemsets

Item	Count
Beer	3
Bread	4
Diapers	4
Milk	4

Candidate
2-itemsets

Itemset	Count
{Beer, Bread}	2
{Beer, Diapers}	3
{Beer, Milk}	2
{Bread, Diapers}	3
{Bread, Milk}	3
{Diapers, Milk}	3

Frequent Itemset Generation

Minimum support count = 3

Candidate 2-itemsets

Itemset	Count
{Beer, Diapers}	3
{Bread, Diapers}	3
{Bread, Milk}	3
{Diapers, Milk}	3

Candidate Large 2-itemsets

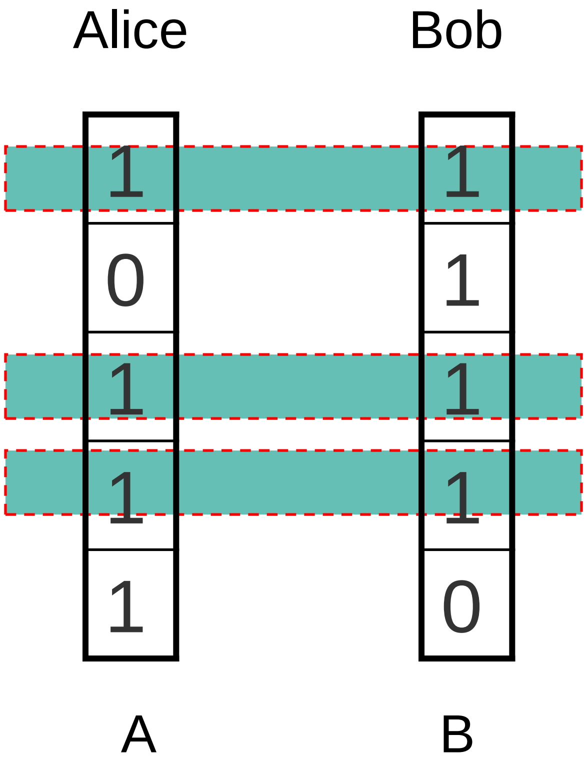
Itemset	Count
{Beer, Diapers}	3
{Bread, Diapers}	3
{Bread, Milk}	3
{Diapers, Milk}	3

REMOVED

An Example

- c.count is the vector product.
- Let’s use A to denote Alice’s attribute vector and B to denote Bob’s attribute vector.
- AB is a candidate frequent itemset, then c.count = A B = 3.

- How to conduct this computation across parties without compromising each party’s data privacy?



Alice	Bob
1	1
0	1
1	1
1	1
1	0
A	B

Homomorphic Encryption

[Paillier 1999]

- Privacy-preserving protocols are based on Homomorphic Encryption.
- Specifically, we use the following additive homomorphism property:

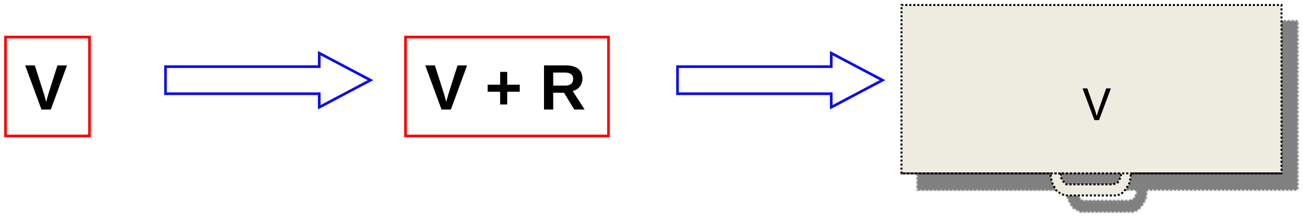
$$e(m_1) \quad e(m_2) \qquad\qquad e(m_n) \quad e(m_1 \quad m_2 \qquad\qquad m_n)$$

- Where e is an encryption function and m_i is the data to be encrypted and $e(m_i) \quad 0$.

Digital Envelope

[Chaum85]

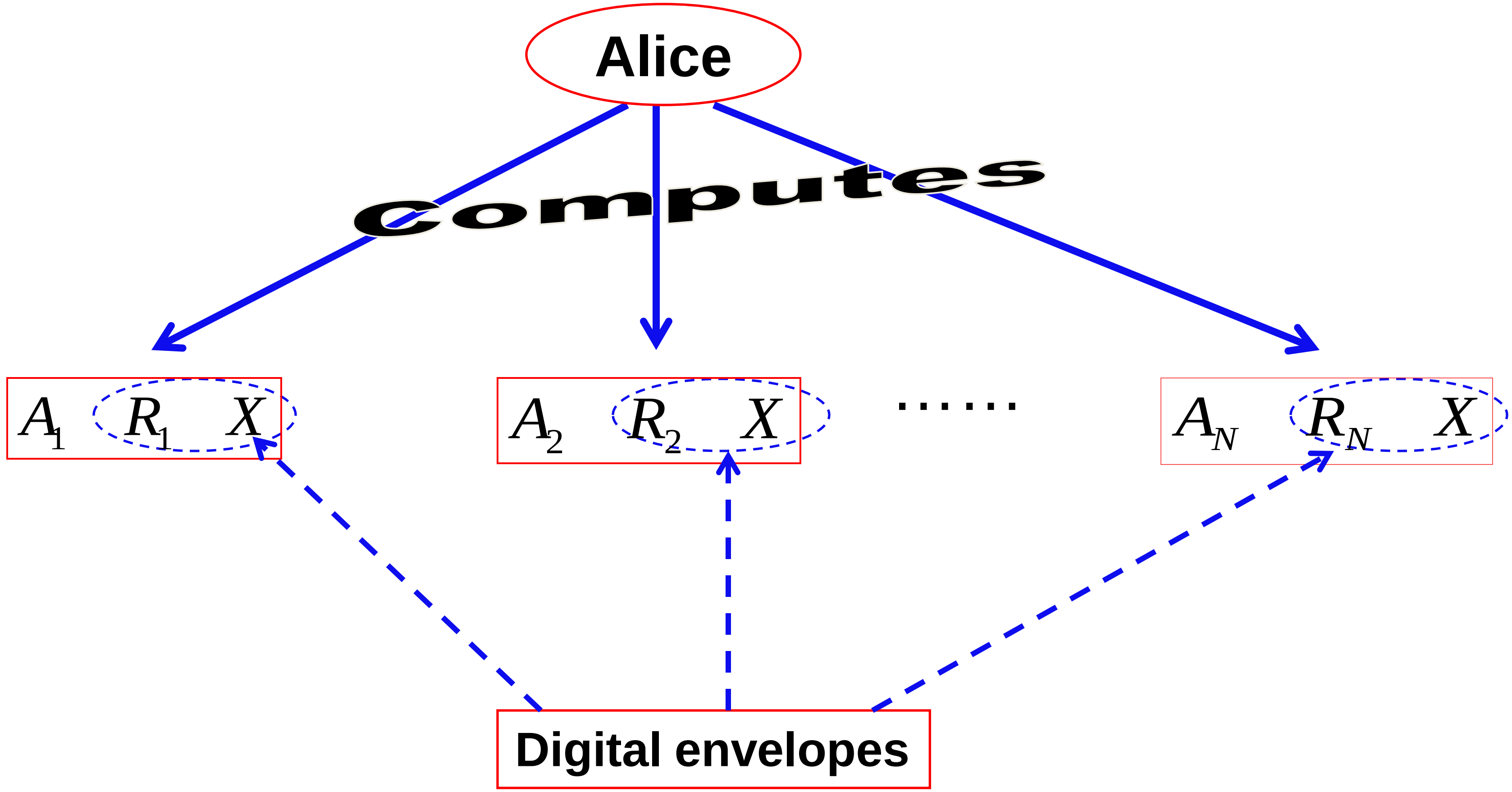
- A digital envelope is a random number (a set of random numbers) only known by the owner of private data.



Frequency Count Protocol

- Assume Alice’s attribute vector is A and Bob’s attribute vector is B.
- Each vector contains N elements.
- A_i : the ith element of A.
- B_i : the ith element of B.
- One of parties is randomly chosen as a key generator, e.g, Alice, who generates (e, d) and an integer X > N. e and X will be shared with Bob.
- Let’s use e(.) to denote encryption and d(.) to denote decryption.

Step 1



Alice

Computes

$A_1 \quad R_1 \quad X$

$A_2 \quad R_2 \quad X$

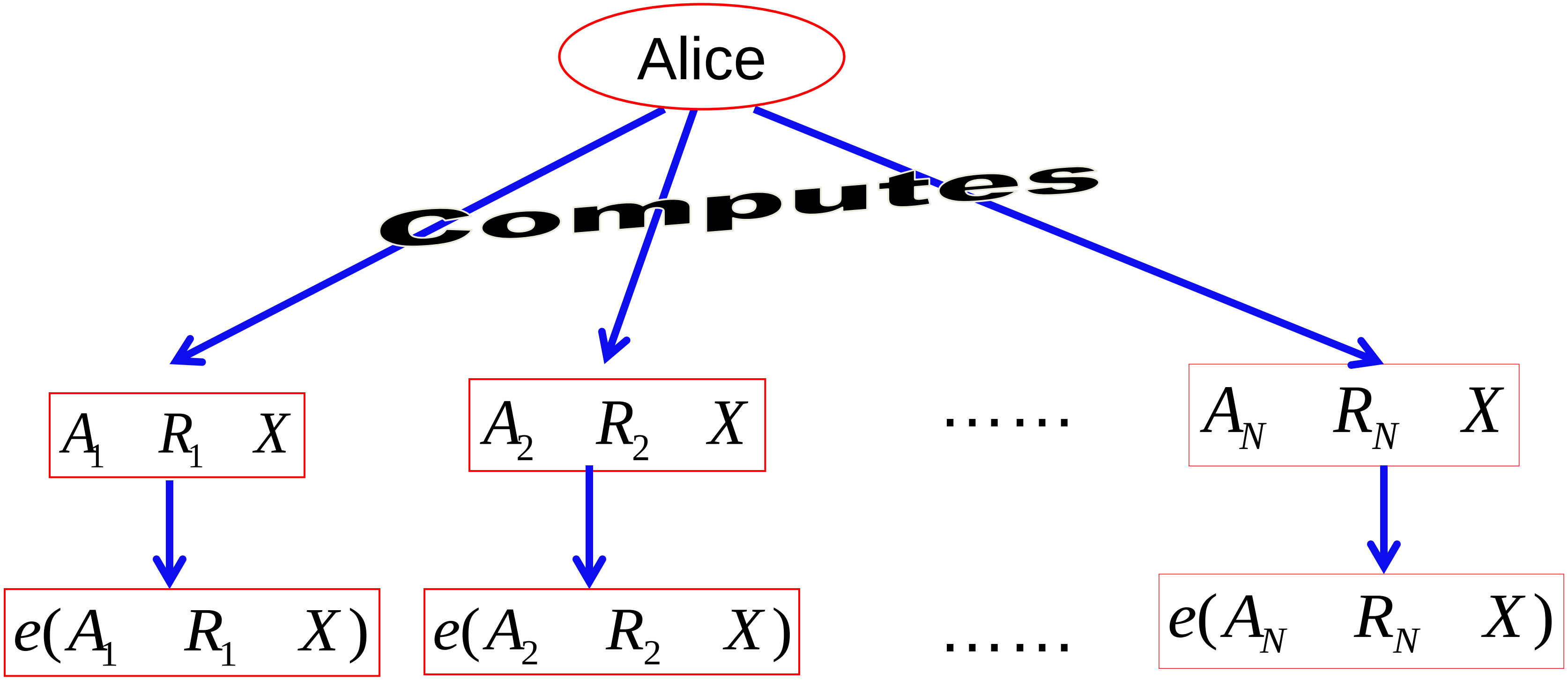
……

$A_N \quad R_N \quad X$

Digital envelopes

$R_1, R_2 \qquad , R_N$ → A set of random integers generated by Alice

Step 1



Alice

Computes

$A_1 \quad R_1 \quad X$

$A_2 \quad R_2 \quad X$

……

$A_N \quad R_N \quad X$

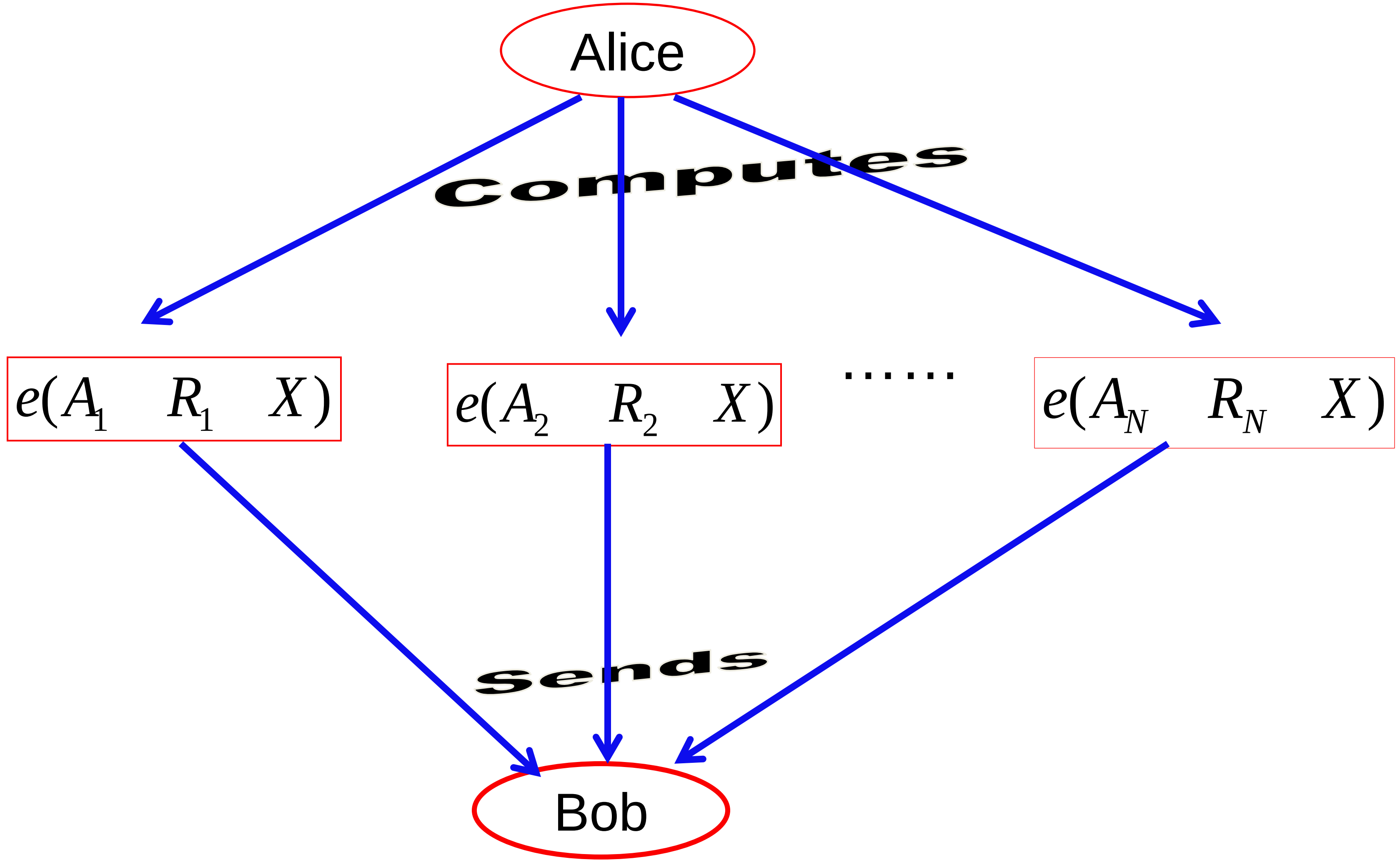
$e(A_1 \quad R_1 \quad X)$

$e(A_2 \quad R_2 \quad X)$

……

$e(A_N \quad R_N \quad X)$

Step 1



Alice

Computes

$e(A_1 \quad R_1 \quad X)$

$e(A_2 \quad R_2 \quad X)$

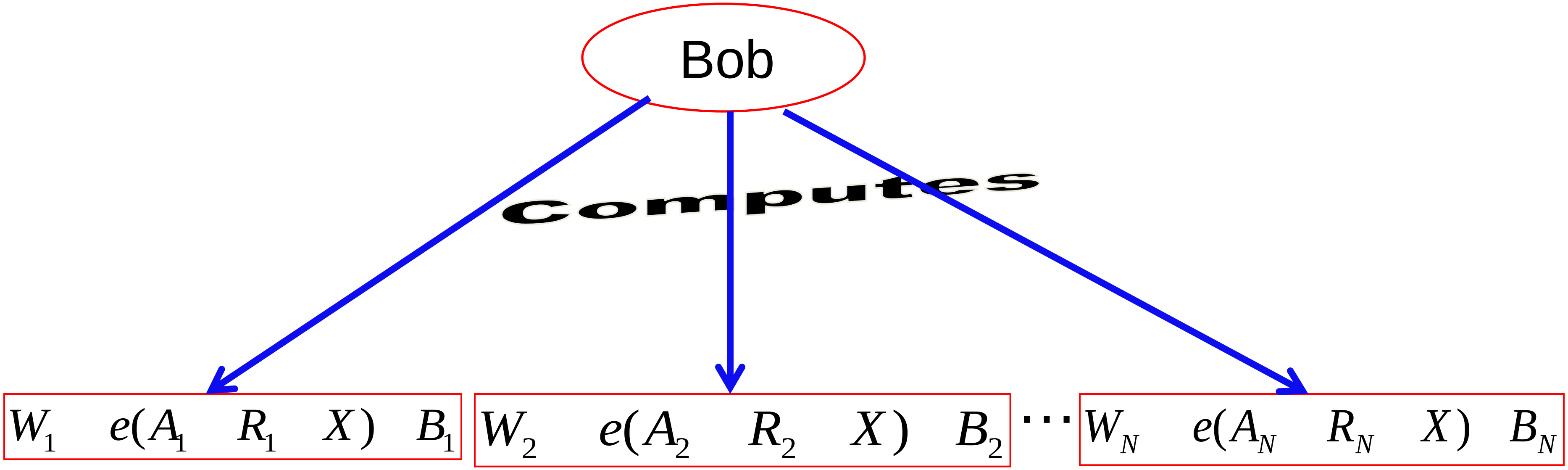
……

$e(A_N \quad R_N \quad X)$

Sends

Bob

Step 2



$W_1 \quad e(A_1 \quad R_1 \quad X) \quad B_1$

$W_2 \quad e(A_2 \quad R_2 \quad X) \quad B_2$

$\cdots$

$W_N \quad e(A_N \quad R_N \quad X) \quad B_N$

$$B_i \quad 0 \qquad W_i \quad 0$$

$$B_i \quad 1 \qquad W_i \quad e(A_i \quad R_i \quad X) \quad B_i \quad e(A_i \quad R_i \quad X)$$

Step 3

- Bob multiplies all the $w_i s$ for those $B_i s$ that are not equal to 0. In other words, Bob computes the multiplication of all non-zero $w_i s$, e.g.,$W \qquad W_i$ where $W_i \quad 0$.

$$\boxed{W \quad W_1 \quad W_2 \qquad\qquad W_j}$$

$$\boxed{W \qquad W_1 \quad W_2 \qquad\qquad W_j}$$

$$[e(A_1 \quad R_1 \quad X) \; \boxed{\begin{matrix} B_1 \\ \| \\ 1 \end{matrix}}\,] \; [e(A_2 \quad R_2 \quad X) \; \boxed{\begin{matrix} B_2 \\ \| \\ 1 \end{matrix}}\,] \qquad [e(A_j \quad R_j \quad X) \; \boxed{\begin{matrix} B_j \\ \| \\ 1 \end{matrix}}\,]$$

$$\boxed{W \quad W_1 \quad W_2 \qquad\qquad W_j}$$

$$\boxed{\quad [e(A_1 \quad R_1 \quad X) \quad 1] \;\; [e(A_2 \quad R_2 \quad X) \quad 1] \qquad\qquad [e(A_j \quad R_j \quad X) \quad 1]}$$

$$W \qquad W_1 \quad W_2 \qquad\qquad W_j$$

$$e(A_1 \quad R_1 \quad X) \quad e(A_2 \quad R_2 \quad X) \qquad\qquad e(A_j \quad R_j \quad X)$$

According to the property of homomorphic encryption

$$e(A_1 \quad A_2 \qquad\qquad A_j \quad (R_1 \quad R_2 \qquad\qquad R_j) \quad X)$$

Step 4

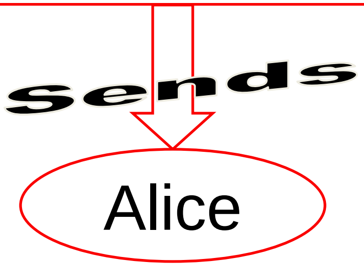
- Bob generates an integer R' .

- Bob then computes

$$\boxed{W' \quad W \quad e(R' \quad X)}$$

According to the property of homomorphic encryption

$$\boxed{e(A_1 \quad A_2 \qquad\qquad A_j \quad (R_1 \quad R_2 \qquad\qquad R_j \quad R') \quad X)}$$



Sends

Alice

The Final Step

- Alice decrypts W' and computes modulo X.
- She then obtains the frequency count.

$$\boxed{c.count \quad d(W') \bmod X}$$

The Final Step

$$c.count$$

$$d(e(A_1 \quad A_2 \qquad\qquad A_j \quad (R_1 \quad R_2 \qquad\qquad R_j \quad R') \quad X)) \bmod X$$

The Final Step

c.count

$$(A_1 \quad A_2 \quad\quad A_j \quad (R_1 \quad R_2 \quad\quad R_j \quad R') \quad X) \bmod X$$

$$(A_1 \quad A_2 \quad\quad A_j) \quad N \quad X$$

$$((R_1 \quad R_2 \quad\quad R_j \quad R') \quad X) \bmod X \quad 0$$

c.count

$$A_1 \quad A_2 \quad\quad A_j$$

Correctness Analysis

$B_i \quad 1$

$A_i \quad 1$

$c.count$

$$c.count \quad A_1 \quad A_2 \qquad\qquad A_j$$

When $B_i \quad 1, (A_1 \quad A_2 \qquad\qquad A_j)$ gives the total number of times that both A_i and B_i are 1s.

Therefore, the frequency count is correctly computed.

Privacy Analysis

Goal: Bob never sees Alice’s data values.

Alice's Privacy

- All the information that Bob obtains from Alice is

$$e(A_1 \quad R_1 \quad X), e(A_2 \quad R_2 \quad X), \quad , e(A_N \quad R_N \quad X) \cdot$$

- Since Bob doesn’t know the decryption key d, he cannot get Alice’s original data values.

Privacy Analysis

Goal: Alice never sees Bob’s data values.

Bob's Privacy

The information that Alice obtains from Bob is $W' \quad e(A_1 \quad A_2 \qquad A_j \quad (R_1 \quad R_2 \qquad R_j \quad R') \quad X)$ for those $B_i \quad 1$.

Alice computes $d(W') \bmod X$. She only obtains the frequency count and cannot know Bob’s original data values.

Complexity Analysis

Communication Cost

Linear in the number of transactions

The total number elements in each attribute vector

$(N \quad 1)$ where N is the total number transactions and   is the number of bits for each encrypted element.

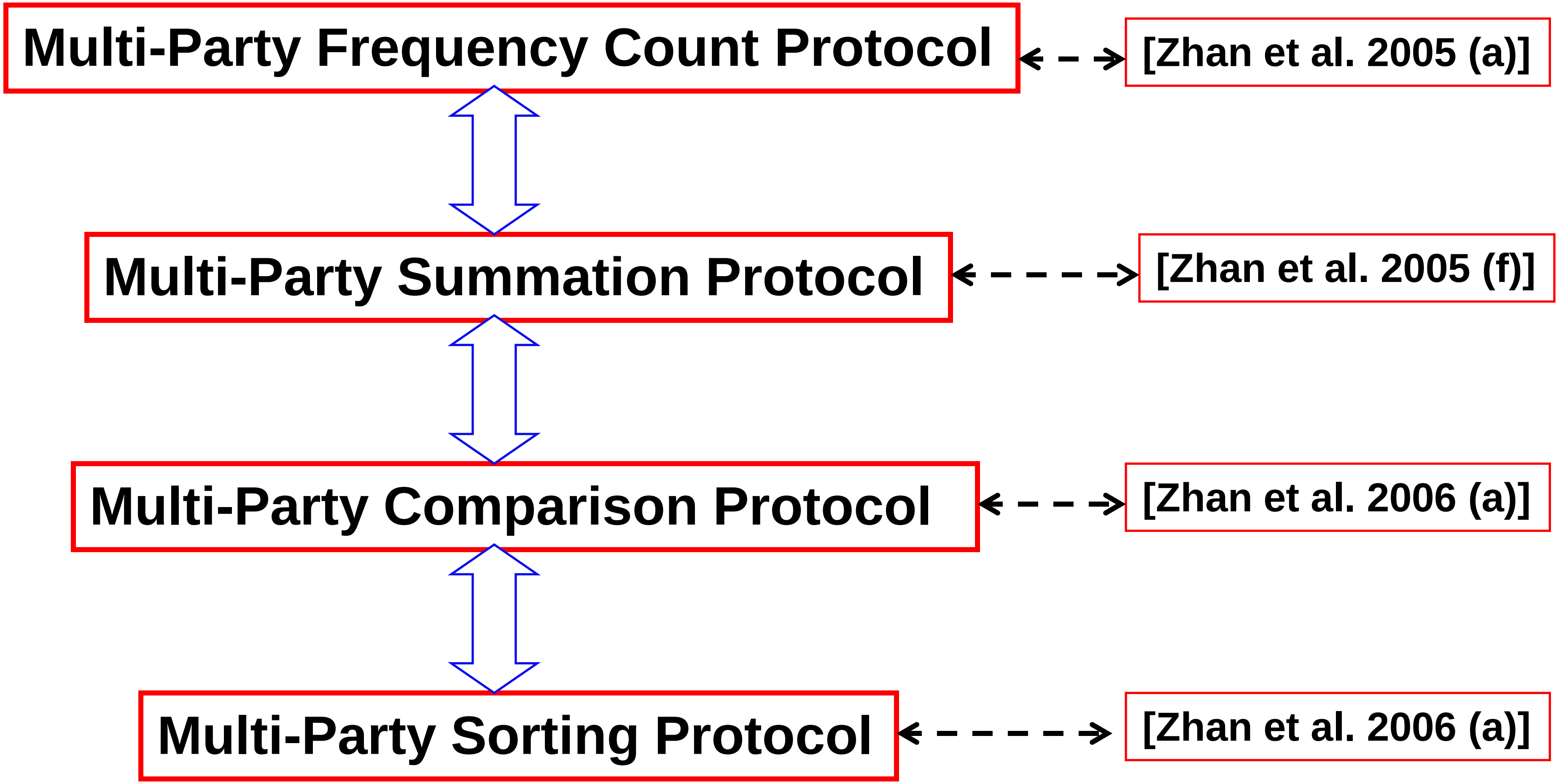
Complexity Analysis

Computation Cost

Linear in the number of transactions

The computational cost is (10N + 20 + g) where N is the total number transactions and g is the computational cost for generating a key pair.

Other Privacy-Oriented Protocols



Multi-Party Frequency Count Protocol

[Zhan et al. 2005 (a)]

Multi-Party Summation Protocol

[Zhan et al. 2005 (f)]

Multi-Party Comparison Protocol

[Zhan et al. 2006 (a)]

Multi-Party Sorting Protocol

[Zhan et al. 2006 (a)]

Our Contributions

- A formal definition of privacy for privacy-preserving collaborative data mining.
- Solutions for data mining tasks for both horizontal collaboration and vertical collaboration.

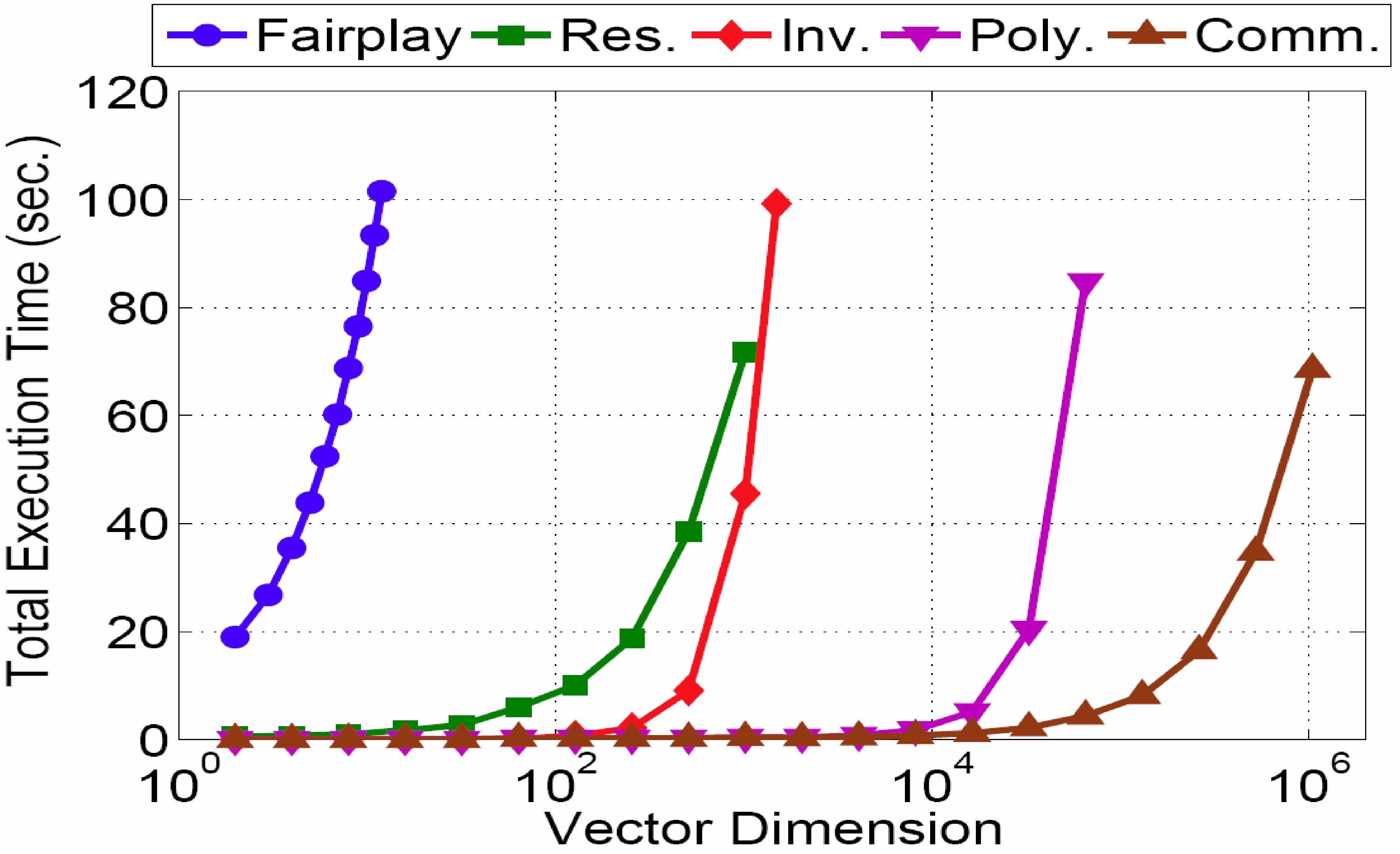
Association Rule Mining [Zhan et.al.2004(a), Zhan et.al. 2004(b)].
Sequential pattern mining [Zhan et. al.2004(c), Zhan et. al. 2005 (a)].
Naïve Bayesian classification [Zhan et. al.2004(d), Zhan et. al.2005 (b)].
Decision tree classification [Zhan et. al. 2005 (a)-(b)].
k-nearest neighbor classification [Zhan et. al. 2005 (c)-(d)].
Support vector machine classification [Zhan et.al. 2008 (e) - (f)].
Clustering [Zhan et. al.2005 (g), Zhan et. al. 2008(a)].

- Simulation with various factors including the number of parties involved in the computation, the encryption key size and the size of data set, etc.

Efficient Privacy-Preserving Collaborative Compiler System Using Scalar Product

Approach Secure	Inv.	Comm.	Poly.	Res.	FP.
Information-theoretically secure					
Computationally secure					

Execution Time



Future Works

- Social Computing (IEEE SocialCom)
- http://www.iisocialcom.org/conference/socialcom2009/

?