Information-Theoretic Co-clustering

Authors: I. S. Dhillon, S. Mallela, and D. S. Modha

.

MALNIS Presentation

Qiufen Qi, Zheyuan Yu

20 May 2004

Outline

1. Introduction
2. Information Theory Concepts
3. Co-Clustering Algorithm
4. Experimental Results
5. Conclusions and Future Work

1. Introduction

Document Clustering: grouping together of ”similar” documents

- Hard Clustering
  - Each document belongs to a single cluster
- Soft Clustering
  - Each document is probabilistically assigned to clusters

One way clustering vs. co-clustering

- One way clustering
  - Clustering documents base on their word distribution
  - Clustering words by their co-occurrence in documents
- Co-clustering
  - Word clustering induces document clustering while document clustering induces word clustering
    * Implicit dimensionality reduction at each step
    * Computationally economical

Co-clustering Methods

- Information-Theoretic Co-clustering

  Co-clustering by finding a pair of maps from rows to row-clusters and from columns to column-clusters, with minimum mutual information loss. Dhillon, et al(2003)

- Bipartite Spectral Graph Partitioning

  Co-clustering by finding minimum cut vertex partitions in a bipartite graph between documents and words. Dhillon, et al(2001)
  Drawback: Each word cluster need to be associated with a document clustering.

2. Information Theory Concepts

Entropy of a random variable X with probability distribution p:

$$H(p) = -\sum_x p(x) \log p(x)$$

- Measure of the average uncertainty

The **Kullback-Leibler(KL) Divergence**:

$$D(p\|q) = \sum_x p(x) \log \frac{p(x)}{q(x)}$$

- Measure of how different two probability distributions are.
- $D(p\|q) \geq 0$; $D(p\|q) = 0$ iff $p = q$
- Not strictly a distance.

Mutual Information between random variables X and Y:

$$I(X;Y) = H(X) - H(X|Y) = \sum_{xy} p(x,y) \log \frac{p(x,y)}{p(x)p(y)}$$

- The amount of information X contains about Y
- Vice versa
- $I(X;Y) = I(Y;X)$;
- $I(X;Y) \geq 0$

"Optimal" Co-Clustering

Finding maps C_x and C_y,

$$C_X : \{x_1, x_2, ..., x_m\} \rightarrow \{\hat{x}_1, \hat{x}_2, ..., \hat{x}_k\}$$
$$C_Y : \{y_1, y_2, ..., y_n\} \rightarrow \{\hat{y}_1, \hat{y}_2, ..., \hat{y}_l\}$$

- **hard-clustering** of the rows and columns
- **loss** in **mutual information** is **minimized**

$$I(X;Y) - I(\hat{X};\hat{Y})$$

Example: $\begin{pmatrix} .05 & .05 & .05 & 0 & 0 & 0 \\ .05 & .05 & .05 & 0 & 0 & 0 \\ 0 & 0 & 0 & .05 & .05 & .05 \\ 0 & 0 & 0 & .05 & .05 & .05 \\ .04 & .04 & 0 & .04 & .04 & .04 \\ .04 & .04 & .04 & 0 & .04 & .04 \end{pmatrix} \rightarrow \begin{pmatrix} .3 & 0 \\ 0 & .3 \\ .2 & .2 \end{pmatrix}$ Loss: 0.0957

Finding optimal Co-Clustering

Lemma: The loss in mutual information can be expressed as the **distance** of p(X, Y) to an **approximation** q(X, Y)

$$I(X;Y) - I(\hat{X};\hat{Y}) = D(p(X,Y) \parallel q(X,Y))$$

q is approximation of p:

$$q(x,y) = p(\hat{x},\hat{y})p(x|\hat{x})p(y|\hat{y}), \text{ where } x \in \hat{x}, y \in \hat{y}.$$

Related to data compression problem

- Transmit the cluster identifies $\hat{X}$ and $\hat{Y}$;
- Transmit X given $\hat{X}$;
- Transmit Y given $\hat{Y}$

Example: Calculating $q(x,y)$: approximation of $p(x,y)$

$$p(x,y) = \begin{pmatrix} .05 & .05 & .05 & 0 & 0 & 0 \\ .05 & .05 & .05 & 0 & 0 & 0 \\ 0 & 0 & 0 & .05 & .05 & .05 \\ 0 & 0 & 0 & .05 & .05 & .05 \\ .04 & .04 & 0 & .04 & .04 & .04 \\ .04 & .04 & .04 & 0 & .04 & .04 \end{pmatrix}$$

Three row clusters:
$\hat{x}_1 = \{x_1, x_2\}, \hat{x}_2 = \{x_3, x_4\}$ and $\hat{x}_3 = \{x_5, x_6\}$

Two column clusters:
$\hat{y}_1 = \{y_1, y_2, y_3\}$ and $\hat{y}_2 = \{y_4, y_5, y_6\}$

$$\underbrace{\begin{pmatrix} .5 & 0 & 0 \\ .5 & 0 & 0 \\ 0 & .5 & 0 \\ 0 & .5 & 0 \\ 0 & 0 & .5 \\ 0 & 0 & .5 \end{pmatrix}}_{p(x|\hat{x})} \underbrace{\begin{pmatrix} .3 & 0 \\ 0 & .3 \\ .2 & .2 \end{pmatrix}}_{p(\hat{x},\hat{y})} \underbrace{\begin{pmatrix} .36 & .36 & .28 & 0 & 0 & 0 \\ 0 & 0 & 0 & .28 & .36 & .36 \end{pmatrix}}_{p(y|\hat{y})} = \underbrace{\begin{pmatrix} .054 & .054 & .042 & 0 & 0 & 0 \\ .054 & .054 & .042 & 0 & 0 & 0 \\ 0 & 0 & 0 & .042 & .054 & .054 \\ 0 & 0 & 0 & .042 & .054 & .054 \\ .036 & .036 & 0.28 & .028 & .036 & .036 \\ .036 & .036 & .028 & .028 & .036 & .036 \end{pmatrix}}_{q(x,y)}$$

Objective function for loss in mutual information

The loss in mutual information can be expressed as

• a weighted sum of relative entropies between row distribution and row cluster distribution

$$D(p(X,Y) \parallel q(X,Y)) = \sum_{\hat{x}} \sum_{x:\in\hat{x}} p(x) D(p(Y|x) \parallel q(Y|\hat{x}))$$

• a weighted sum of relative entropies between column distribution and column cluster distribution

$$D(p(X,Y) \parallel q(X,Y)) = \sum_{\hat{y}} \sum_{y\in\hat{y}} p(y) D(p(X|y) \parallel q(X|\hat{y}))$$

It allows us to define:

• a row cluster prototype: $q(Y|\hat{x})$

• a column cluster prototype: $q(X|\hat{y})$

Lead to a "natural" algorithm

3. Co-Clustering Algorithm

- Input:

$p(X,Y)$ - the joint probability distribution
k - the desired number of row clusters
l - the desired number of column clusters

- Output:

The partition function C_X and C_Y

1. Inititalization: Set $t = 0$. Start with some initial partition functions $C_X^{(0)}$ and $C_Y^{(0)}$. Compute $q^{(0)}(\hat{X},\hat{Y})$ and the distribution for each row-cluster prototype $q^{(0)}(Y|\hat{x})$, $1 \leq \hat{x} \leq k$
2. Row re-clustering: For each row x, assign it to the "closet" row-cluster prototype. Update $C_X^{(t+1)}$. $C_Y^{(t+1)} = C_Y^{(t)}$.
3. Computer $q^{(t+1)}(\hat{X},\hat{Y})$ and the distribution for each column-cluster prototype $q^{(t+1)}(X|\hat{y})$, $1 \leq \hat{y} \leq l$

Co-clustering Algorithm (con't)

4. Column re-clustering: For each column y, assign it to the "closest" column-cluster prototype. Update $C_Y^{(t+2)}$. $C_X^{(t+2)} = C_X^{(t+1)}$.
5. Compute $q^{(t+2)}(\hat{X}, \hat{Y})$ and the distribution for each row cluster prototype $q^{(t+2)}(Y|\hat{x})$.
6. If the change of the loss in mutual information, i.e.

$$D(p(X,Y)||q^t(X,Y)) - D(p(X,Y) \parallel q^{t+2}(X,Y))$$

is small (say 10^{-3}), Return $C_X^{(t+2)}$ and $C_Y^{(t+2)}$; Else set $t = t+2$ and go to step 2.

Properties of Co-clustering Algorithm

- Co-clustering monotonically decreases loss in mutual information
- Co-clustering converges to a local minimum
- Can be generalized to multi-dimensional contingency tables
- Implicit dimensionality reduction at each step helps overcome sparsity & high-dimensionality
- Computationally efficient: $O(nt(k+l))$, where

n is the number of non zeros in the input joint distribution
t is the number of iterations
k is the desired number of row clusters
l is the desired number of column clusters

4. Experimental Results

- Algorithms to be compared
- Data sets
- Evaluation measures
- Results and discussion

Algorithms to be compared:

- IB-double: Information Bottleneck Double Clustering
- IDC: Iterative Double Clustering
- 1D-clustering: without any word clustering (information theoretic method?)

Data sets:

- NG20: Binary, Multi5 and Multi10 (with and without subjects, 500 documents each)
- SMART: CLASSIC3 - MEDLINE (1033), CISI (1460) and CRANFIELD (1400)
- Top 2000 words were selected by mutual information (frequency ?) after the stop words were removed.

Evaluation Measures:

- **Confusion matrix**:

Each entry(i, j) represents the number of documents in the cluster i that belong to true class j

- **Micro-averaged-precision**:

$$p = \frac{\sum_1^l a_i}{N}$$

where:

a_i - the number of correctly assigned documents in cluster
l - the number of document clusters (classes)
N - the total number of documents in a whole data set

Results

Co-clustering			1D-clustering		
992	4	8	**944**	9	98
40	**1452**	7	71	**1431**	5
1	4	**1387**	18	20	**1297**

Table 3: Co-clustering accurately recovers original clusters in the *CLASSIC3* data set.

Binary				Binary_subject			
Co-clustering		1D-clustering		Co-clustering		1D-clustering	
244	4	**178**	104	**241**	11	**179**	94
6	**246**	72	**146**	9	**239**	71	**156**

Table 4: Co-clustering obtains better clustering results compared to one dimensional document clustering on Binary and Binary_subject data sets

Table 3: Co-clustering (0.9835), 1D-clustering (0.9432)
Table 4: Co-clustering (0.98, 0.96), 1D-clustering(0.67, 0.648)

Results (con't)

- Co-clustering performs much better than IB-Double and 1D-clustering and is comparable with IDC

- Word clustering can alleviate the problem of clustering in high dimensions

	Co-clustering	1D-clustering	IB-Double	IDC
Binary	0.98	0.64	0.70	
Binary_subject	0.96	0.67		0.85
Multi5	0.87	0.34	0.5	
Multi5_subject	0.89	0.37		0.88
Multi10	0.56	0.17	0.35	
Multi10_subject	0.54	0.19		0.55

(Note: The peak values were selected.)

Results (con't)

- Different data sets achieve their maximum at different number of word clusters

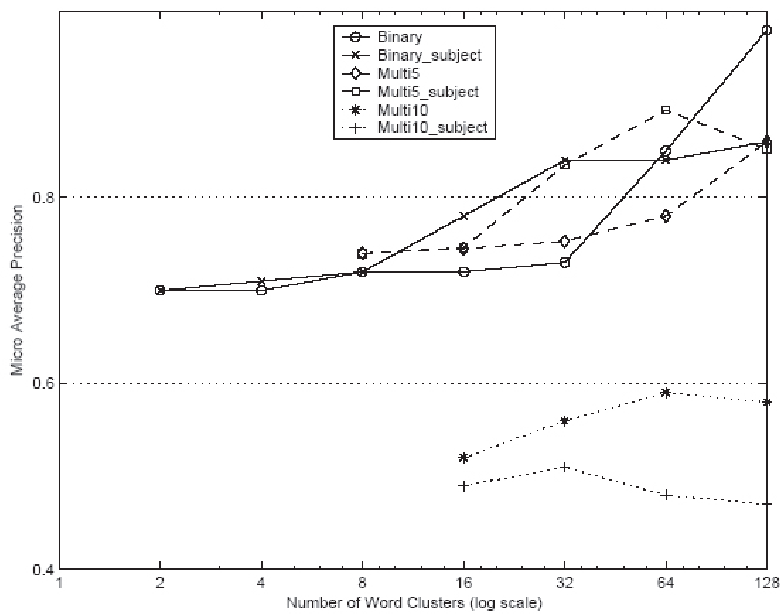

Figure 2: Micro-averaged-precision values with varied number of word clusters using co-clustering on different *NG20* data sets.

Results (con't)

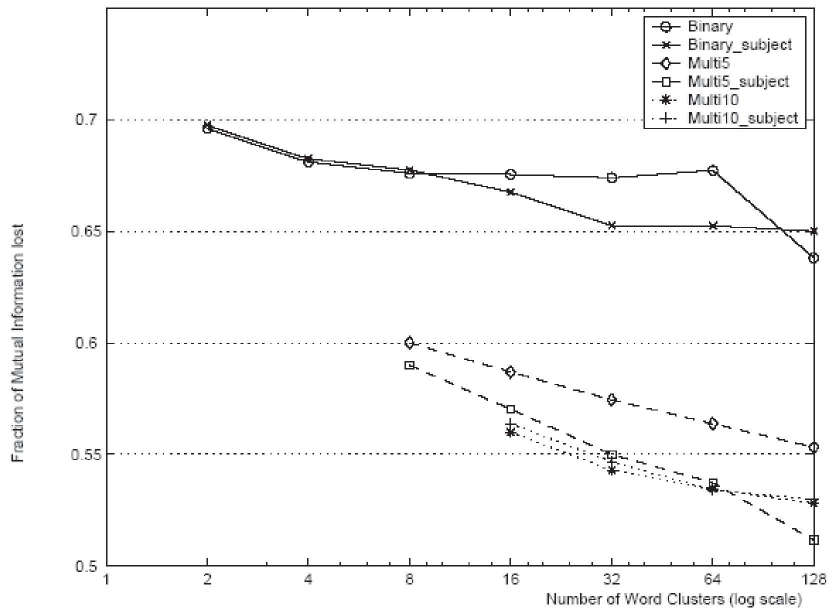

Figure 3: Fraction of mutual information lost with varied number of word clusters using co-clustering on different *NG20* data sets.

Correlation between Figure 2 & 3: the lower the loss in mutual information, the better is the clustering

Results (con't)

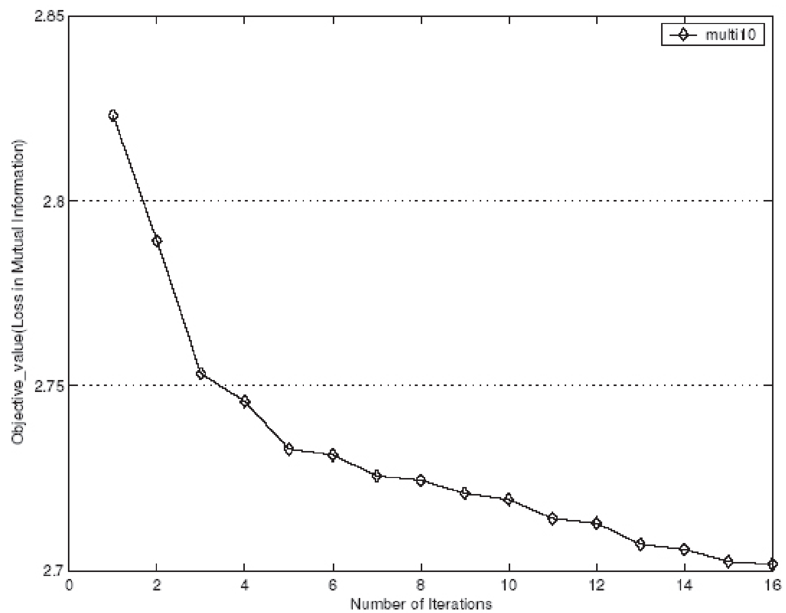

Figure 4: Loss in mutual information decreases monotonically with the number of iterations on a typical co-clustering run on the Multi10 data set.

Co-clustering converges quickly in about 20 iterations on all the tested datasets

Results (con't)

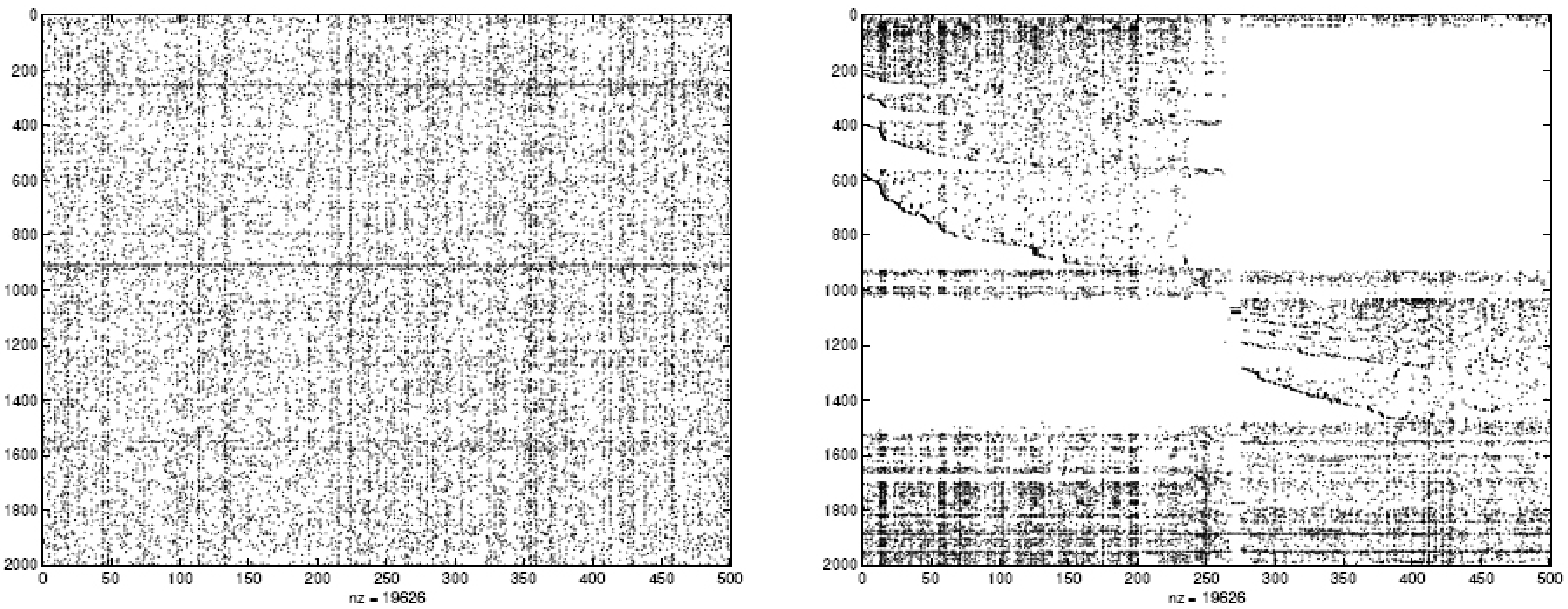

Figure 5: Sparsity structure of the Binary_subject word-document co-occurrence matrix before(left) and after(right) co-clustering reveals the underlying structure of various co-clusters (2 document clusters and 100 word clusters). The shaded regions represent the non-zero entries.

5. Conclusions and Future Work

- Information-theoretic approach to co-clustering

- Implicit dimensionality reduction at each step to overcome sparsity & high-dimensionality

- Theoretical approach has the potential of extending to other problems:

- Multi-dimensional co-clustering
- MDL (Minimum Description Length) to choose number of co-clusters
- Generalize co-clustering to an abstract multivariate clustering setting

REFERENCES

1. Information-Theoretic Co-clustering
I. S. Dhillon, S. Mallela, and D. S. Modha Proceedings of The Ninth ACM SIGKDD Conference on Knowledge Discovery and Data Mining(KDD), pages 89-98, August, 2003.