

Document Clustering Using Word Clusters via the Information Bottleneck Method

Noam Slonim and Naftali Tishby

Conference on Research and Development
in Information Retrieval (SIGIR)

2000

Presented by Bret Ehlert
May 14, 2002

Document Clustering

Top: Computers: Artificial Intelligence (1,752)

[Description](#)

- [Academic Departments](#) (47)
- [Agents](#) (75)
- [Applications](#) (20)
- [Artificial Life@](#) (302)
- [Associations](#) (14)
- [Belief Networks](#) (47)
- [Companies](#) (67)
- [Conferences and Events](#) (67)
- [Creativity](#) (44)
- [Data Mining@](#) (195)
- [Distributed Projects](#) (13)
- [Fuzzy](#) (59)
- [Games](#) (14)
- [Genetic Programming](#) (52)
- [Knowledge Representation](#) (55)
- [Machine Learning](#) (205)
- [Natural Language](#) (288)
- [Neural Networks](#) (293)
- [People](#) (51)
- [Philosophy](#) (36)
- [Programming Languages](#) (1)
- [Publications](#) (29)
- [Qualitative Physics](#) (2)
- [Robotics](#) (190)
- [Vision](#) (59)

See also:

- [Science: Social Sciences: Cognitive Science](#) (350)
- [Science: Social Sciences: Psychology](#) (2,868)
- [Science: Technology: Cybernetics](#) (155)

- Document clustering is closely related to text classification.

Traditional Clustering Methods

- Represent a document as a vector of weights for the terms that occur in the document.

	w_1	w_2	w_3	...	w_{124080}	w_{124081}	...	word _n
doc ₁ :	0.0	0.75	0.0	...	0.0	0.13	...	0.0
doc ₂ :	0.6	0.21	0.0	...	0.36	0.0	...	0.0

- This representation has many disadvantages:
 - High dimensionality
 - Sparseness
 - Loss of word ordering information
- Clustering documents using the distances between pairs of vectors is troublesome.
 - The Information Bottleneck is an alternative method that does not rely on vector distances.

Dimensionality Reduction

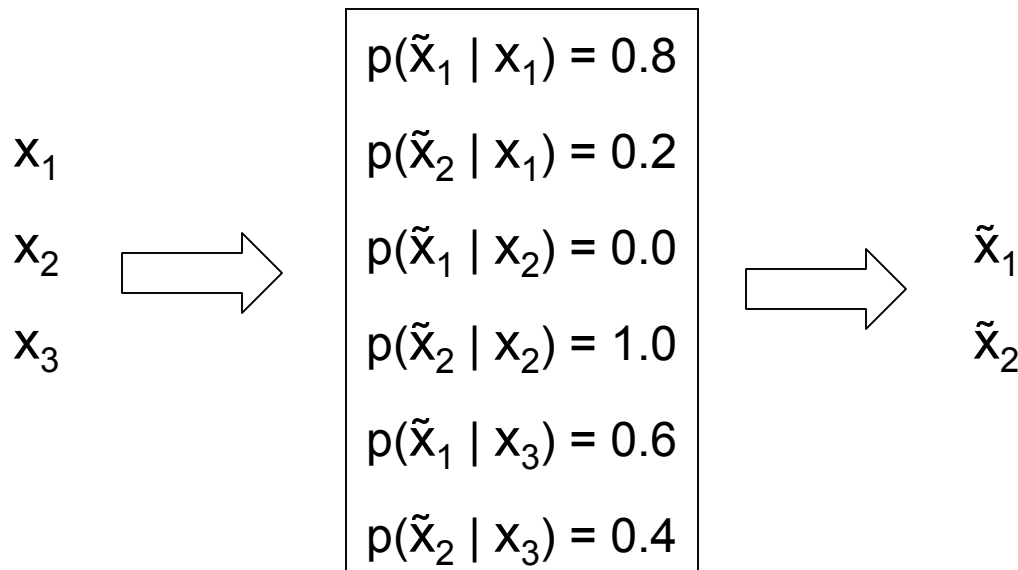
- Dimensionality reduction is beneficial for improved accuracy and efficiency when clustering documents.
 - Latent semantic indexing (LSI)
 - Information Gain and Mutual Information Measures
 - Chi-Squared Statistic
 - Term Strength Algorithm
 - **Distributional Clustering**
 - Cluster words based on their distribution across documents
 - The Information Bottleneck is a distributional clustering method

The Information Bottleneck

- A distributional clustering method
 - Used to cluster words, reducing the dimensionality of document representations.
 - Used to cluster documents.
- The agglomerative algorithm presented in the paper is a special case of a general approach:
 - Tishby, Pereira, and Bialek. *The Information Bottleneck Method*. 37-th Annual Allerton Conference on Communication. 1999.

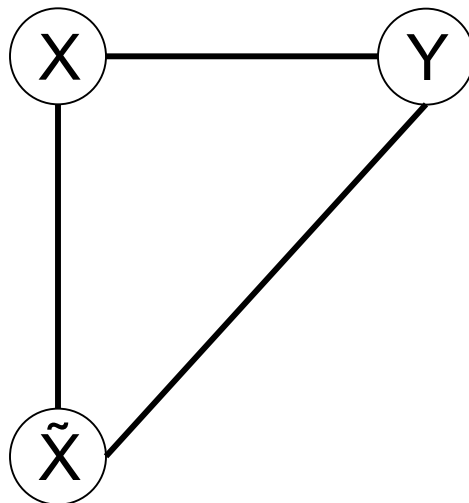
The Information Bottleneck: $X \rightarrow \tilde{X}$

- Find a mapping between $x \in X$ and $\tilde{x} \in \tilde{X}$, characterized by a conditional probability distribution $p(\tilde{x} | x)$.
 - For example, if X is the set of words, \tilde{X} is a new representation of words where $|\tilde{X}| < |X|$.
- This mapping induces a soft partitioning of X : each $x \in X$ maps to $\tilde{x} \in \tilde{X}$ with probability $p(\tilde{x} | x)$.



The Information Bottleneck: $Y \rightarrow X \rightarrow \tilde{X}$

- Suppose the variable X is an observation of Y , where Y is the variable of interest.
 - $x \in X$ is evidence concerning the outcome $y \in Y$
 - For example, $x \in X$ is a word and $y \in Y$ is a document
- We want the mapping from $x \in X$ to $\tilde{x} \in \tilde{X}$ to preserve as much information about Y as possible.



Entropy

- **Entropy** measures the uncertainty about a discrete random variable X :

$$H(X) = - \sum_{x \in X} p(X = x) \log_2 p(X = x)$$

- Entropy defines the lower bound on the number of bits needed to accurately encode X .
- **Conditional entropy** of X given Y describes the amount of remaining uncertainty about X after an observation of Y :

$$H(X | Y) = E[H(X | y)] = - \sum_x \sum_y p(x, y) \log p(x | y)$$

- **Relative entropy**, or *Kullback-Leibler (KL) distance*, measures the distance between two distributions:

$$D_{\text{KL}}(p, q) = \sum_x p(x) \log \frac{p(x)}{q(x)}$$

Mutual Information

- The ***Mutual Information*** of X and Y measures the amount of uncertainty about X that is resolved by observing Y :

$$I(X, Y) = H(X) - H(X | Y)$$

$$I(X, Y) = \sum_x \sum_y p(x, y) \log \left(\frac{p(x, y)}{p(x)p(y)} \right)$$

- This is also the relative entropy between the joint distribution of X and Y and the product of the distributions of X and Y .
- Note that $I(X, Y) = I(Y, X)$

Information Theory Examples

Examples using two different joint probability distributions

	X_1	X_2
y_1	0.25	0.25
y_2	0.25	0.25

$$H(X) = -0.5 \log 0.5 - 0.5 \log 0.5 = \mathbf{1.0}$$

$$H(Y) = \mathbf{1.0}$$

$$H(X|Y) = -0.25 \log 0.5 - 0.25 \log 0.5 - \\ 0.25 \log 0.5 - 0.25 \log 0.5 \\ = \mathbf{1.0}$$

$$H(Y|X) = \mathbf{1.0}$$

$$I(X,Y) = H(X) - H(X|Y) = 1.0 - 1.0 = \mathbf{0}$$

	X_1	X_2
y_1	0.75	0.01
y_2	0.05	0.19

$$H(X) = -0.8 \log 0.8 - 0.2 \log 0.2 = \mathbf{0.72}$$

$$H(Y) = -0.76 \log 0.76 - 0.24 \log 0.24 = \mathbf{0.79}$$

$$H(X|Y) = -0.75 \log 0.98 - 0.01 \log 0.01 - \\ 0.05 \log 0.21 - 0.19 \log 0.79 \\ = \mathbf{0.25}$$

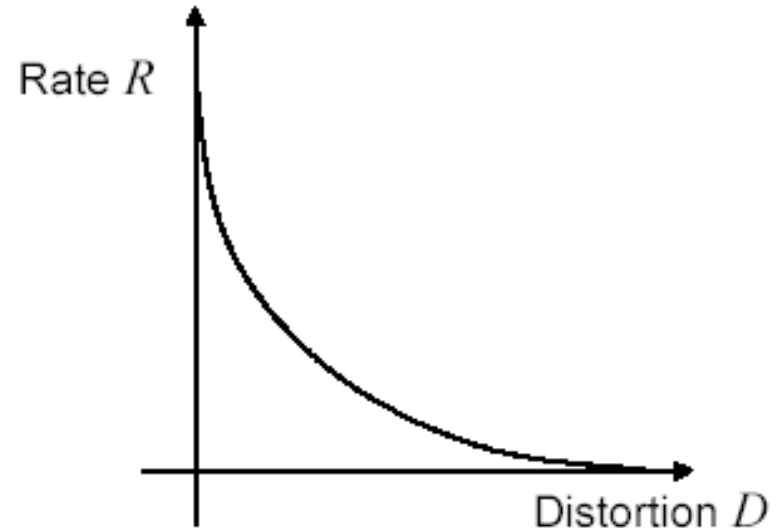
$$H(Y|X) = -0.75 \log 0.94 - 0.01 \log 0.05 - \\ 0.05 \log 0.06 - 0.19 \log 0.95 \\ = \mathbf{0.32}$$

$$I(X,Y) = H(X) - H(X|Y) = 0.72 - 0.25 = \mathbf{0.47}$$

$$I(X,Y) = H(Y) - H(Y|X) = 0.79 - 0.32 = \mathbf{0.47}$$

Lossy Compression

- We want to transmit signal X using a compressed version \tilde{X} .
 - Lower the bit-rate R (compression) by sending \tilde{X} with some acceptable distortion D .



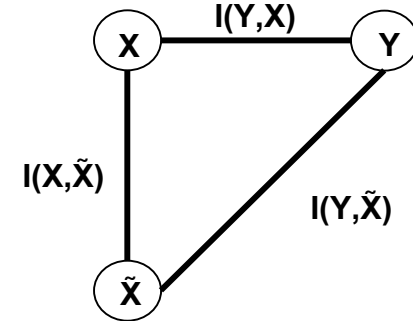
- Two compression problems:
 - Given a maximum rate R , minimize distortion D
 - Given an acceptable amount of distortion D , minimize rate R
- Solve simultaneously using an unconstrained Lagrangian cost function:

$$J = D + \lambda R$$

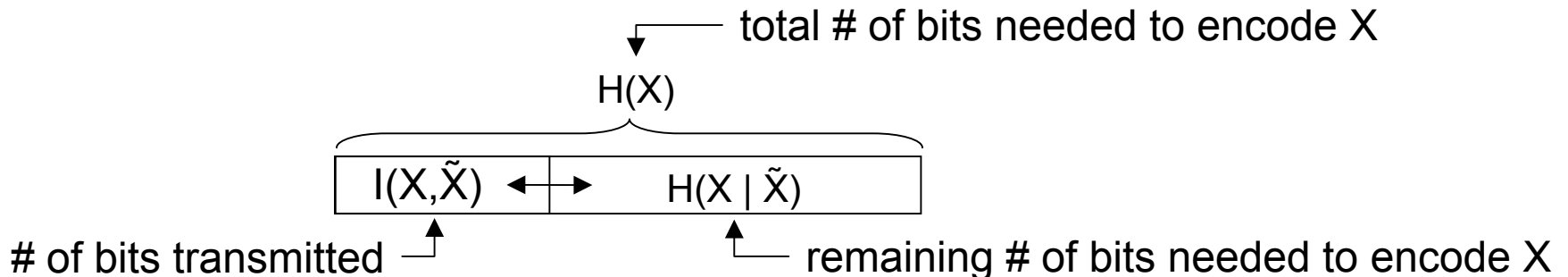
- The Information Bottleneck method measures distortion using $I(Y, \tilde{X})$ and compression using $I(X, \tilde{X})$.

Rate and Distortion Using Mutual Information

- Maximizing $I(Y, \tilde{X})$ minimizes distortion.
 - Recall that X is an observation of the variable of interest, Y .
 - $I(Y, \tilde{X})$ measures the quality of the compression.



- Minimizing $I(X, \tilde{X})$ maximizes compression.
 - Recall, $I(X, \tilde{X}) = H(X) - H(X | \tilde{X})$
 - $H(X)$ is constant, so minimizing $I(X, \tilde{X})$ maximizes $H(X | \tilde{X})$
 - $H(X | \tilde{X})$ defines the minimum additional number of bits needed on average to represent $x \in X$ after observing $\tilde{x} \in \tilde{X}$



The Information Bottleneck Cost Function

Minimize the Lagrangian cost function:

$$L = \underbrace{I(X, \tilde{X})}_{\text{minimize}} - \beta \underbrace{I(Y, \tilde{X})}_{\text{maximize}}$$

- β is the tradeoff between maximizing compression and minimizing distortion.
 - If $\beta = 0$, every $x \in X$ maps to every $\tilde{x} \in \tilde{X}$ with the same probability
 - No information is being transmitted
 - In the limit $\beta \rightarrow \infty$, $I(Y, \tilde{X})$ is maximized, given the capacity of \tilde{X} .
 - This results in hard clustering
- Note, to find a solution we must fix the cardinality of \tilde{X} .

Solving the Cost Function

Minimize the Lagrangian cost function:

$$L = \underbrace{I(X, \tilde{X})}_{\text{minimize}} - \beta \underbrace{I(Y, \tilde{X})}_{\text{maximize}}$$

- Remember, we want to find a mapping between $x \in X$ and $\tilde{x} \in \tilde{X}$, characterized by a conditional probability distribution $p(\tilde{x} | x)$.

Solve $\frac{\partial L}{\partial p(\tilde{x} | x)} = 0$ for $p(\tilde{x} | x)$.

The Solution

$$p(\tilde{x} | x) = \frac{p(\tilde{x}) \exp \left[-\beta \sum_y p(y | x) \log \frac{p(y | x)}{p(y | \tilde{x})} \right]}{\sum_{\tilde{x}} p(\tilde{x}) \exp \left[-\beta \sum_y p(y | x) \log \frac{p(y | x)}{p(y | \tilde{x})} \right]}$$

$$p(\tilde{x} | x) = \frac{p(\tilde{x})}{\underbrace{Z(x, \beta)}_{\substack{\text{normalization} \\ \text{over all } \tilde{x}}}} \exp \left[- \overset{\text{tradeoff}}{\beta} \underbrace{D_{\text{KL}}(p(y | x), p(y | \tilde{x}))}_{\substack{\text{relative entropy} \\ \text{between } p(y|x) \text{ and } p(y|\tilde{x})}} \right]$$

Equations for $p(\tilde{x})$ and $p(y | \tilde{x})$

$$p(\tilde{x} | x) = \frac{p(\tilde{x})}{Z(x, \beta)} \exp \left[-\beta D_{\text{KL}} (p(y | x), p(y | \tilde{x})) \right]$$

We also need equations for $p(\tilde{x})$ and $p(y | \tilde{x})$

$$p(\tilde{x}) = \sum_x p(\tilde{x} | x) p(x)$$

$$\begin{aligned} p(y | \tilde{x}) &= \frac{p(\tilde{x} | y) p(y)}{p(\tilde{x})} \\ &= \frac{p(y)}{p(\tilde{x})} \sum_x p(\tilde{x} | x) p(x | y) \\ &= \frac{1}{p(\tilde{x})} \sum_x p(\tilde{x} | x) p(y | x) p(x) \end{aligned}$$

Information Bottleneck Iterative Algorithm

- Alternating iterations are run to solve the three equations:

$$p(\tilde{x} | x) = \frac{p(\tilde{x})}{Z(x, \beta)} \exp \left[-\beta D_{\text{KL}} (p(y | x), p(y | \tilde{x})) \right]$$

$$p(\tilde{x}) = \sum_x p(\tilde{x} | x) p(x)$$

$$p(y | \tilde{x}) = \frac{1}{p(\tilde{x})} \sum_x p(\tilde{x} | x) p(y | x) p(x)$$

- Convergence can be proven.
- Separate solutions exist for different values of β and cardinalities of \tilde{X} , which are fixed for the algorithm.
 - Use deterministic annealing to search for best β and cardinality of \tilde{X} .

Hard Clustering Limit

- Taking the limit $\beta \rightarrow \infty$ results in maximizing $I(Y, \tilde{X})$ with no constraint on compression.
- In the limit $\beta \rightarrow \infty$, $p(\tilde{x}, x) = 0$ or 1 . Hence, each $x \in X$ will map to exactly one $\tilde{x} \in \tilde{X}$

Recall, $I(Y, \tilde{X}) = H(Y) - H(Y | \tilde{X})$

$H(Y)$ is constant, so maximizing $I(Y, \tilde{X})$ minimizes $H(Y | \tilde{X})$

where

$$H(Y | \tilde{X}) = - \sum_{\tilde{x}} \sum_y p(\tilde{x}, y) \log p(\tilde{x} | y)$$

$$H(Y | \tilde{X}) = - \sum_{\tilde{x}} \sum_y p(\tilde{x} | y) p(y) \log p(\tilde{x} | y)$$

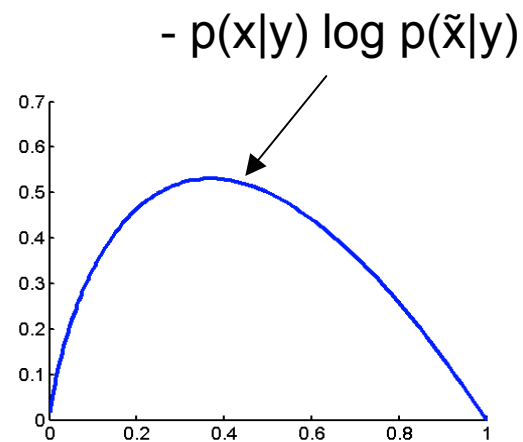
Each element of $H(Y | \tilde{X})$ is minimized when $p(\tilde{x} | y) = 0$ or 1

$$p(\tilde{x} | y) = \sum_x p(\tilde{x} | x) p(x | y) = p(x_1 | y) p(\tilde{x} | x_1) + p(x_2 | y) p(\tilde{x} | x_2) + \dots + p(x_n | y) p(\tilde{x} | x_n)$$

where $\sum_x p(\tilde{x} | x) = 1$

This is closest to 1 when $p(\tilde{x}, x) = 1$ for the largest $p(x_i | y)$

This is closest to 0 when $p(\tilde{x}, x) = 1$ for the smallest $p(x_i | y)$



New Equations for the Hard Clustering Limit

- Think of each $\tilde{x} \in \tilde{X}$ as a set of elements $x \in X$

$$p(\tilde{x} | x) = \begin{cases} 1 & \text{if } x \in \tilde{x} \\ 0 & \text{otherwise} \end{cases}$$

$$p(\tilde{x}) = \sum_{x \in \tilde{x}} p(x)$$

$$p(y | \tilde{x}) = \frac{1}{p(\tilde{x})} \sum_{x \in \tilde{x}} p(y | x) p(x)$$

- These equations define the properties of the optimal distribution but not the optimal distribution itself.

Agglomerative Algorithm

- Start with a partition of $|X|$ singleton clusters. $|\tilde{X}| = |X|$
- At each step, decrease $|\tilde{X}|$, merging two clusters that locally minimize the loss of mutual information $I(Y, \tilde{X})$.
- Merging \tilde{x}_i and \tilde{x}_j into x_* : $(\tilde{x}_i, \tilde{x}_j) \Rightarrow \tilde{x}_*$

$$p(\tilde{x}_* | x) = \begin{cases} 1 & \text{if } x \in \tilde{x}_i \text{ or } x \in \tilde{x}_j \\ 0 & \text{otherwise} \end{cases}$$

$$p(y | \tilde{x}_*) = \frac{p(\tilde{x}_i)}{p(\tilde{x}_*)} p(y | \tilde{x}_i) + \frac{p(\tilde{x}_j)}{p(\tilde{x}_*)} p(y | \tilde{x}_j)$$

$$p(\tilde{x}_*) = p(\tilde{x}_i) + p(\tilde{x}_j)$$

- Merge the pair \tilde{x}_i and \tilde{x}_j that minimizes the decrease in $I(Y, \tilde{X})$, defined as

$$\partial I(\tilde{x}_i, \tilde{x}_j) = I(\tilde{X}_{\text{before}}, Y) - I(\tilde{X}_{\text{after}}, Y)$$

With a Little Algebra...

$$\begin{aligned}\partial I(\tilde{\mathbf{X}}_i, \tilde{\mathbf{X}}_j) &= I(\tilde{\mathbf{X}}_{\text{before}}, Y) - I(\tilde{\mathbf{X}}_{\text{after}}, Y) \\ &= \sum_{\tilde{\mathbf{x}} \in \mathbf{X}_{\text{before}}} \sum_y p(\tilde{\mathbf{x}}, y) \log \frac{p(\tilde{\mathbf{x}}, y)}{p(\tilde{\mathbf{x}})p(y)} - \sum_{\tilde{\mathbf{x}} \in \mathbf{X}_{\text{after}}} \sum_y p(\tilde{\mathbf{x}}, y) \log \frac{p(\tilde{\mathbf{x}}, y)}{p(\tilde{\mathbf{x}})p(y)} \\ &= \left[\sum_y p(\tilde{\mathbf{x}}_i, y) \log \frac{p(\tilde{\mathbf{x}}_i, y)}{p(\tilde{\mathbf{x}}_i)p(y)} + \sum_y p(\tilde{\mathbf{x}}_j, y) \log \frac{p(\tilde{\mathbf{x}}_j, y)}{p(\tilde{\mathbf{x}}_j)p(y)} \right] - \sum_y p(\tilde{\mathbf{x}}_*, y) \log \frac{p(\tilde{\mathbf{x}}_*, y)}{p(\tilde{\mathbf{x}}_*)p(y)} \\ &= p(\tilde{\mathbf{x}}_i) \underbrace{\sum_y p(y | \tilde{\mathbf{x}}_i) \log \frac{p(y | \tilde{\mathbf{x}}_i)}{p(y | \tilde{\mathbf{x}}_*)}}_{D_{\text{KL}}[p(y|\tilde{\mathbf{x}}_i), p(y|\tilde{\mathbf{x}}_*)]} + p(\tilde{\mathbf{x}}_j) \underbrace{\sum_y p(y | \tilde{\mathbf{x}}_j) \log \frac{p(y | \tilde{\mathbf{x}}_j)}{p(y | \tilde{\mathbf{x}}_*)}}_{D_{\text{KL}}[p(y|\tilde{\mathbf{x}}_j), p(y|\tilde{\mathbf{x}}_*)]}\end{aligned}$$

which is in the form a Jensen-Shannon (JS) divergence

$$D_{\text{JS}}(p_i, p_j) = \pi_i D_{\text{KL}}(p_i, p_*) + \pi_j D_{\text{KL}}(p_j, p_*)$$

Complexity of the Agglomerative Algorithm

- $O(|X|)$ iterations are required to check all sizes of \tilde{X} .
- Each iteration requires calculating $\delta I(\tilde{x}_i, \tilde{x}_j)$ for each possible pair \tilde{x}_i, \tilde{x}_j .
 - $O(|x|^2)$ possible pairs
 - Each calculation requires $O(|Y|)$ time
- Total running time is $O(|X|^3 |Y|)$
 - Can be reduced to $O(|X|^2 |Y|)$ by reusing $\delta I(\tilde{x}_i, \tilde{x}_j)$ calculations.

- The agglomerative algorithm does not guarantee a globally optimal solution.
 - $I(Y, \tilde{X})$ is minimized locally for each merge.

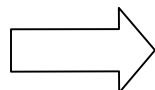
Double Clustering Procedure

- First obtain word-clusters to represent documents in a reduced dimensional space.
- Then cluster the documents using the word-cluster representation.

Double Clustering Example

Given a joint probability distribution $p(x, y)$:

	x_1	x_2	x_3	x_4
y_1	0.11	0.12	0.02	0.01
y_2	0.09	0.10	0.04	0.03
y_3	0.02	0.01	0.13	0.16
y_4	0.03	0.02	0.06	0.05



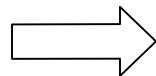
Find \tilde{X} :

$x_1, x_2 \in \tilde{X}_1$

$x_3, x_4 \in \tilde{X}_2$

Represent documents using $p(\tilde{x}, y)$:

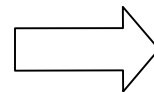
	\tilde{x}_1	\tilde{x}_2
y_1	0.23	0.03
y_2	0.19	0.07
y_3	0.03	0.29
y_4	0.05	0.11



Find \tilde{Y} :

$y_1, y_2 \in \tilde{y}_1$

$y_3, y_4 \in \tilde{y}_2$



	\tilde{x}_1	\tilde{x}_2
\tilde{y}_1	0.42	0.10
\tilde{y}_2	0.08	0.40

Alternative Clustering Methods

- L1 norm estimation of the JS-divergence

$$L_1(p, q) = \sum_y |p(y) - q(y)|$$

$$d_{i,j} = (p(\tilde{x}_i) + p(\tilde{x}_j)) L_1(p(y | \tilde{x}_i), p(y | \tilde{x}_j))$$

- Ward's method as an alternative to the JS-divergence

$$d_{i,j} = \frac{p(\tilde{x}_i) p(\tilde{x}_j)}{p(\tilde{x}_i) + p(\tilde{x}_j)} \sum_y (p(y | \tilde{x}_i) - p(y | \tilde{x}_j))^2$$

- Complete-linkage algorithm
 - Merge the most similar documents
 - Similarity is measured as the cosine of the angle between tf-idf vector representations of documents

Experimental Data

- Measure accuracy by comparing resulting clusters to real document categories.
- 20Newsgroups corpus (Lang 1995)
 - 20,000 articles distributed among 20 UseNet discussion groups
- 10 subsets were used, each containing randomly selected documents:

Dataset	Newsgroups included	Documents per group	Total documents
Science	sci.crypt, sci.electronics, sci.med, sci.space	500	2000
Binary _{1,2,3}	talk.politics.mideast, talk.politics.misc	250	500
Muti5 _{1,2,3}	comp.graphics, rec.motorcycles, rec.sport.baseball, sci.space, talk.politics.mideast	100	500
Muti10 _{1,2,3}	alt.atheism, comp.sys.mac.hardware, misc.foresale, rec.autos, rec.sport.hockey, sci.crypt, sci.electronics, sci.med, sci.space, talk.politics.gun	50	500

Experimental Setup

- Each data set was represented using 2000 words.
 - The 2000 words with the highest mutual information between words and documents were selected.
- Tested the following algorithms:
 - IB-double - Information bottleneck double clustering procedure
 - IB-single - Information bottleneck, without clustering words
 - L1-double
 - L1-single
 - Ward-double
 - Ward-single
 - Complete tf-idf
- Tested performance using 10, 20, 30, 40, and 50 word-clusters.
- The number of document clusters was identical to the number of real clusters.

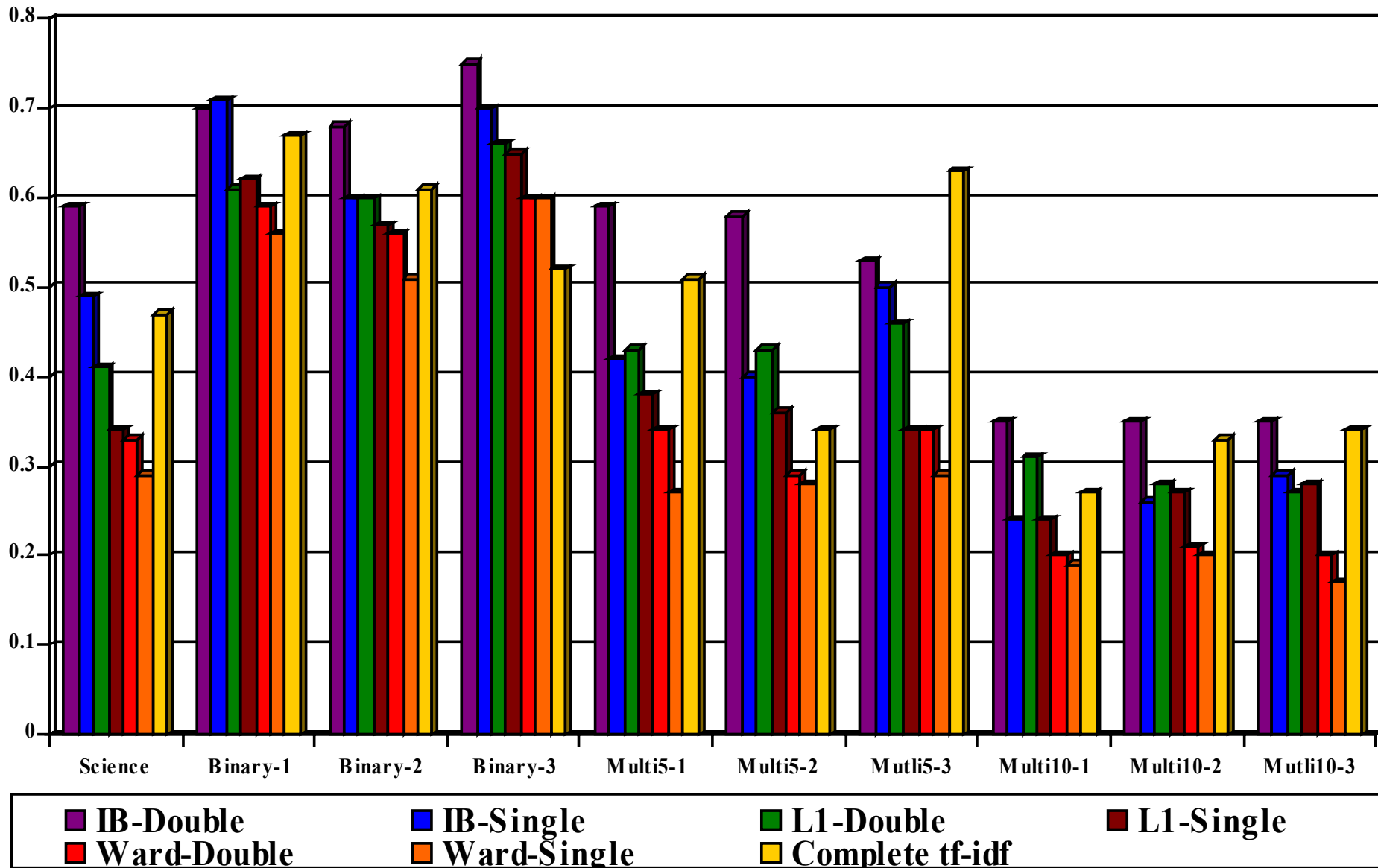
Measuring Accuracy

Contingency table for Muti5₂ data set
using the IB-double with 10 word-clusters:

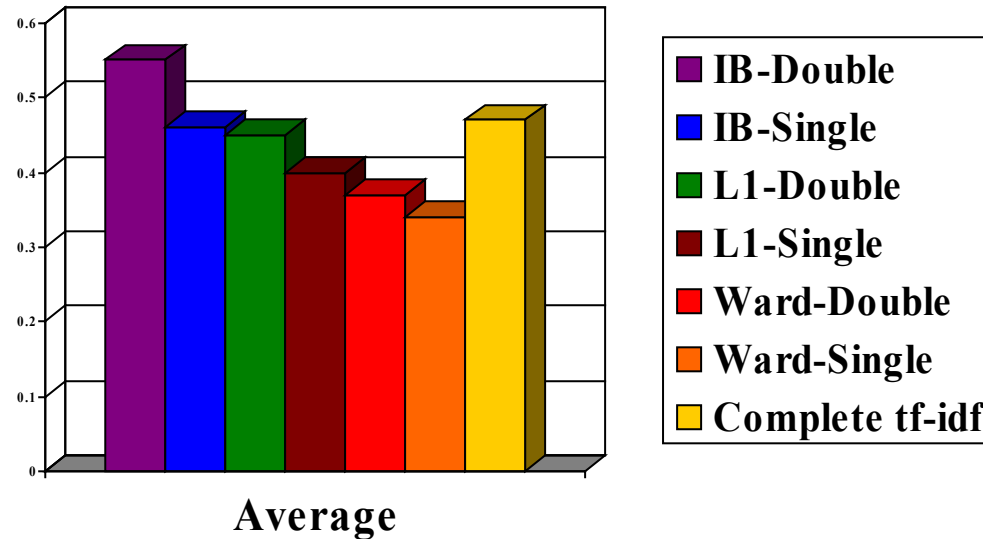
	<i>graphics</i>	<i>motorcycles</i>	<i>baseball</i>	<i>space</i>	<i>mideast</i>
$x_1^?$	78/100	3	11	6	10
$x_2^?$	3	68/100	7	5	5
$x_3^?$	4	5	59/100	8	9
$x_4^?$	6	14	13	68/100	13
$x_5^?$	9	10	10	13	63/100

$$\text{Total Accuracy} = \frac{\frac{78}{100} + \frac{68}{100} + \frac{59}{100} + \frac{68}{100} + \frac{63}{100}}{5} = 0.67$$

Experimental Results



Summary of Results



- IB-double gives the best overall performance
 - The IB measure had better performance than all other double clustering methods.
- In 46 out of 50 trials, double-clustering improved performance for all distance measures.
- All algorithms performed best on the binary data sets, and weakest on the Multi10 data sets.
 - Increasing the number of clusters results in decreasing performance

Discussion

- The agglomerative algorithm does not use the general solution found for Lagrangian optimization.
 - It does not guarantee a globally optimal solution
 - It has a running time of $O(|X|^3 |Y|)$.
- Further work may look at simultaneously clustering words and documents.
- The Information Bottleneck method has also been applied to clustering words for text classification.
 - Slonim & Tishby. *The Power of Word Clusters for Text Classification*. 2001