A General Model for Clustering Binary Data

Tao Li
School of Computer Science
Florida International University
Miami, FL 33199
taoli@cs.fiu.edu

ABSTRACT

Clustering is the problem of identifying the distribution of patterns and intrinsic correlations in large data sets by partitioning the data points into similarity classes. This paper studies the problem of clustering binary data. This is the case for market basket datasets where the transactions contain items and for document datasets where the documents contain "bag of words". The contribution of the paper is three-fold. First a general binary data clustering model is presented. The model treats the data and features equally, based on their symmetric association relations, and explicitly describes the data assignments as well as feature assignments. We characterize several variations with different optimization procedures for the general model. Second, we also establish the connections between our clustering model with other existing clustering methods. Third, we also discuss the problem for determining the number of clusters for binary clustering. Experimental results show the effectiveness of the proposed clustering model.

Categories and Subject Descriptors

I.2 [**Artificial Intelligence**]: Learning; I.5.3 [**Pattern Recognition**]: Clustering

General Terms

Algorithms, Experimentation, Theory

Keywords

Clustering, Binary Data, Matrix Approximation, General Model

1. INTRODUCTION

The problem of clustering data arises in many disciplines and has a wide range of applications. Intuitively, clustering is the problem of partitioning a finite set of points in a multi-dimensional space into classes (called clusters) so that (i) the points belonging to the same class are *similar* and (ii) the points belonging to different classes are *dissimilar*. The clustering problem has been studied extensively in machine learning, databases, and statistics from various perspectives and with various approaches and focuses.

In this paper, we focus our attention on binary datasets. Binary data have been occupying a special place in the domain of data analysis. Typical applications for binary data clustering include market basket data clustering and document clustering. For market basket data, each data transaction can be represented as a binary vector where each element indicates whether or not any of the corresponding item/product was purchased. For document clustering, each document can be represented as a binary vector where each element indicates whether a given word/term was present or not.

The first contribution of the paper is the introduction of a general model for binary clustering. A distinctive characteristic of the binary data is that the features (attributes) they include have the same nature as the data they intend to account for: both are binary. The characteristic suggests a new clustering model where the data and features are treated equally. The new clustering model, explicitly describes the data assignments (assigning data points into clusters) as well as feature assignments (assigning features into clusters). The clustering problem is then formulated as a matrix approximation problem where the clustering objective is to minimize the approximation error between the original data matrix and the reconstructed matrix based on the cluster structures. In general, the approximation can be solved via an iterative alternating least-squares optimization procedure. The optimization procedure simultaneously performs two tasks: data reduction (assigning data points into clusters) and feature identification (identifying features associated with each cluster). By explicit feature assignments, the clustering model produces interpretable descriptions of the resulting clusters. In addition, by iterative feature identification, the clustering model performs an implicit adaptive feature selection at each iteration and flexibly measures the distances between data points. Therefore it works well for high-dimensional data [18]. For many cases, there is usually a symmetric association relations between the data and features in binary clustering: if the set of data points is associated to the set of features, then the set of attributes is associated to the set of data points and vice versa. This symmetric association motivates a block diagonal variant of the general model.

The second contribution of this paper is the presentation of a unified view for binary data clustering. In particular, we show that our new clustering model provides a general framework for binary data clustering based on matrix approximation. Many previously known clustering algorithms can be viewed as different variations derived from the general framework with different constraints and relaxations. Thus our general model provides an elegant basis to establish the connections between various methods while highlighting their differences. In addition, we also examine the relations between our clustering model with other binary clustering models. As a third contribution, we examine the problem of determining the number of clusters with our binary clustering model.

The rest of the paper is organized as follows: Section 2 introduces the general clustering model and describes the general optimization procedure, Section 3 presents the block diagonal variants

KDD'05, August 21–24, 2005, Chicago, Illinois, USA.

of the general model, Section 4 provides the unified view on binary clustering. Section 5 discusses the method for deciding the number of clusters; Section 6 shows our experimental results. Section 7 surveys the related work. Finally, Section 8 presents our conclusions.

2. A GENERAL CLUSTERING MODEL FOR BINARY DATA

2.1 Notations and Formalization

The notations used in the paper are introduced in Table 1.

$W = (w_{ij})_{n\times m}$	The binary data set
$D = (d_1, d_2, \cdots, d_n)$	Set of data points
$F = (f_1, f_2, \cdots, f_m)$	Set of features
K	Number of clusters for data points
C	Number of clusters for features
$P = \{P_1, P_2, \cdots, P_K\}$	Partition of D into K clusters
$i \in P_k, 1 \le k \le K$	i-th data point in cluster P_k
$p_1, p_2, \cdots, p_K$	Sizes for the K data clusters
$Q = \{Q_1, Q_2, \cdots, Q_C\}$	Partition of F into C clusters
$q_1, q_2, \cdots, q_C$	Sizes for the C feature clusters
$j \in Q_c, 1 \le c \le C$	j-th feature in cluster Q_c
$A = (a_{ik})_{n\times K}$	Matrix designating the data membership
$B = (b_{jc})_{m\times C}$	Matrix designating the feature membership
$X = (x_{kc})_{K\times C}$	Matrix specifies/indicates the association between data and features or the cluster representation
Trace(M)	Trace of the Matrix M

Table 1: Notations used throughout the paper.

We first present a general model for binary clustering problem [1]. The model is formally specified as follows:

$$W = AXB^T + E \tag{1}$$

where matrix E denotes the error component. The first term AXB^T characterizes the information of W that can be described by the cluster structures. A and B explicitly designate the cluster memberships for data points and features, respectively. X specifies cluster representation. Let $\hat{W}$ denote the approximation AXB^T and the goal of clustering is to minimize the approximation error (or *sum-of-squared-error*)

$$\begin{aligned} O(A,X,B) &= \| W - \hat{W} \|_F^2 = \text{Trace}[(W-\hat{W})(W-\hat{W})^T] \\ &= \sum_{i=1}^{n}\sum_{j=1}^{m}(w_{ij}-\hat{w}_{ij})^2 \qquad (2) \\ &= \sum_{i=1}^{n}\sum_{j=1}^{m}(w_{ij}-\sum_{k=1}^{K}\sum_{c=1}^{C}a_{ik}b_{jc}x_{kc})^2 \qquad (3) \end{aligned}$$

Note that the Frobenius norm, $\| M \|_F$, of a matrix $M = (M_{ij})$ is given by $\| M \|_F = \sqrt{\sum_{i,j} M_{ij}^2}$.

2.2 General Optimization Procedure

In general, the model leads to the formulation of two-side clustering, i.e., the problem of simultaneously clustering both data points (rows) and features (columns) of a data matrix [7, 17].

Suppose $A = (a_{ik}), a_{ik} \in \{0,1\}, \sum_{k=1}^{K} a_{ik} = 1$, and $B = (b_{jc}), b_{jc} \in \{0,1\}, \sum_{c=1}^{C} b_{jc} = 1$ (i.e., A and B denote the data and feature memberships, respectively). Thus, based on Equation 3, we obtain

$$\begin{aligned} O(A,X,B) &= \| W - \hat{W} \|_F^2 \\ &= \sum_{i=1}^{n}\sum_{j=1}^{m}(w_{ij}-\sum_{k=1}^{K}\sum_{c=1}^{C}a_{ik}b_{jc}x_{kc})^2 \\ &= \sum_{k=1}^{K}\sum_{c=1}^{C}\sum_{i\in P_k}\sum_{j\in Q_c}(w_{ij}-x_{kc})^2 \qquad (4) \end{aligned}$$

For fixed P_k and Q_c, it is easy to check that the optimum X is obtained by

$$x_{kc} = \frac{1}{p_k q_c}\sum_{i\in P_k}\sum_{j\in Q_c} w_{ij} \tag{5}$$

In other words, X can be thought as the matrix of centroids for the two-side clustering problem and it represents the associations between the data clusters and the feature clusters [6]. $O(A,X,B)$ can then be minimized via an iterative procedure of the following steps

1. Given X and B, then the feature partition Q is fixed, $O(A,X,B)$ is minimized by

$$\hat{a}_{ik} = \begin{cases} 1 & \text{if } \sum_{c=1}^{C}\sum_{j\in Q_c}(w_{ij}-x_{kj})^2 < \sum_{c=1}^{C}\sum_{j\in Q_c}(w_{ij}-x_{lj})^2 \\ & \text{for } l = 1,\cdots,K, l \ne k \\ 0 & \text{Otherwise} \end{cases} \tag{6}$$

2. Similarly, Given X and A, then the data partition P is fixed, $O(A,X,B)$ is minimized by

$$\hat{b}_{jc} = \begin{cases} 1 & \text{if } \sum_{k=1}^{K}\sum_{i\in P_k}(w_{ij}-x_{ic})^2 < \sum_{k=1}^{K}\sum_{i\in P_k}(w_{ij}-x_{il})^2 \\ & \text{for } l = 1,\cdots,C, l \ne c \\ 0 & \text{Otherwise} \end{cases} \tag{7}$$

3. Given A and B, X can be computed using Equation 5.

This is a natural extensions of the K-means type algorithm for two-side case [3, 6, 27]. The clustering procedure is shown in Algorithm 1.

Algorithm 1 General Clustering Procedure

Input: ($W_{n\times m}$, K and C)
Output: A: cluster assignment;
B: feature assignment;
begin
1 Initialize A and B;
2 Compute X based on Equation 5.
3. **Iteration:** Do while the stop criterion is not met
begin
3.1 Update A based on Equation 6
3.2 Update B based on Equation 7
2.3 Compute X based on Equation 5
end
3. Output A and B
end

3. BLOCK DIAGONAL CLUSTERING

As mentioned in Section 2, in general, X represents the associations between the data clusters and the feature clusters. For binary data clustering, in many cases, there is usually a symmetric association relations between the data and features: if the set of data

[1] It should be noted that: although the clustering model presented here is motivated from the characteristics of binary data, the model can be generalized to other data types as well.

points is associated to the set of features, then the set of attributes is associated to the set of data points and vice versa. This symmetric association motivates a variant of the general model where X is an identity matrix. Then in the general model, we have $C = K$, i.e., both data points and features have the same number of clusters. The assumption also implies that, after appropriate permutation of the rows and columns, the approximation data take the form of a block diagonal matrix [15].

In this case, AB^T can then be interpreted as the approximation of the original data W. The goal of clustering is then to find a (A,B) that minimizes the squared error between W and its approximation AB^T.

$$O(A,B) = ||W - AB^T||_F^2, \tag{8}$$

3.1 Optimization Procedure

The objective criterion can be expressed as

$$\begin{aligned} O(A,B) &= ||W - AB^T||_F^2 \\ &= \sum_{i=1}^{n}\sum_{j=1}^{m}\left(w_{ij} - \sum_{k=1}^{K} a_{ik}b_{kj}\right)^2 \\ &= \sum_{i=1}^{n}\sum_{k=1}^{K} a_{ik}\sum_{j=1}^{m}(w_{ij} - b_{kj})^2 \\ &= \sum_{i=1}^{n}\sum_{k=1}^{K} a_{ik}\sum_{j=1}^{m}(w_{ij} - y_{kj})^2 \\ &\quad + \sum_{k=1}^{K} n_k \sum_{j=1}^{m}(y_{kj} - b_{kj})^2, \end{aligned} \tag{9}$$

where $y_{kj} = \frac{1}{n_k}\sum_{i=1}^{n} a_{ik}w_{ij}$ and $n_k = \sum_{i=1}^{n} a_{ik}$ (note that we use b_{kj} to denote the entry of B^T.). The objective function can be minimized via an alternating least-squares procedure by alternatively optimizing one of A or B while fixing the other.

Given an estimate of B, new least-squares estimates of the entries of A can be determined by assigning each data point to the closest cluster as follows:

$$\hat{a}_{ik} = \begin{cases} 1 & \text{if } \sum_{j=1}^{m}(w_{ij} - b_{kj})^2 < \sum_{j=1}^{m}(w_{ij} - b_{lj})^2 \\ & \text{for } l = 1,\cdots,K, l \neq k \\ 0 & \text{Otherwise} \end{cases} \tag{10}$$

When A is fixed, $O_{A,B}$ can be minimized with respect to B by minimizing the second part of Equation 9:

$$O'(B) = \sum_{k=1}^{K} n_k \sum_{j=1}^{m}(y_{kj} - b_{kj})^2.$$

Note that y_{kj} can be thought of as the probability that the j-th feature is present in the k-th cluster. Since each b_{kj} is binary [2], i.e., either 0 or 1, $O'(B)$ is minimized by:

$$\hat{b}_{kj} = \begin{cases} 1 & \text{if } y_{kj} > 1/2 \\ 0 & \text{Otherwise} \end{cases} \tag{11}$$

In practice, if a feature has similar association to all clusters, then it is viewed as an outlier at the current stage. The optimization procedure for minimizing Equation 9 alternates between updating A based on Equation 10 and assigning features using Equation 11. After each iteration, we compute the value of the objective criterion $O(A,B)$. If the value is decreased, we then repeat the process; otherwise, the process has arrived at a local minimum. Since the procedure monotonically decreases the objective criterion, it converges to a local optimum. The clustering procedure is shown in Algorithm 2. A preliminary report of the block diagonal clustering is presented in [25].

[2]If the entries of B are arbitrary, then the optimization here can be performed via singular value decomposition.

Algorithm 2 Block Diagonal Clustering Procedure

Input: (data points: $W_{n\times m}$, # of classes: K)
Output: A: cluster assignment;
B: feature assignment;

begin
1. **Initialization:**
1.1 Initialize A
1.2 Compute B based on Equation 11
1.3 Compute $O_0 = O(A,B)$
2. **Iteration:**
begin
2.1 Update A given B (via Equation 10)
2.2 Compute B given A (via Equation 11)
2.3 Compute the value of $O_1 = O(A,B)$;
2.4 if $O_1 < O_0$
2.4.1 $O_0 = O_1$
2.4.2 Repeat from 2.1
2.5 else
2.5.1 break; (Converges)
end
3. **Return** A, B;
end

4. A UNIFIED VIEW BINARY DATA CLUSTERING

In this section, we present a unified view for binary data clustering. In particular, we show that our new clustering model provides a general framework for binary data clustering based on matrix approximation and illustrate several variations that can be derived from the general model in Section 4.1, and examine the relations among between our clustering model with other binary clustering models in Section 4.2. The discussions are summarized in Figure 1.

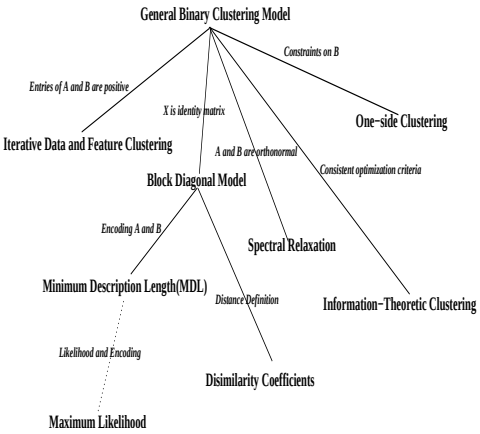


Figure 1: A Unified View on Binary Clustering. The lines represent relations. Note that the relations between maximum likelihood principle and minimum description length (MDL), shown as the dotted line, are well-known facts in machine learning literature.

4.1 Variations of the General Model

4.1.1 One-Side K-means Clustering

Consider the case when $C = m$, then each feature is a cluster by itself and $B = I_{m\times m}$. The model thus reduces to popular one-side clustering, i.e., grouping the data points into clusters. Here we only discuss the one-side clustering for data points. It should be note that, similarly, we can derive one-side feature clustering when $K = n, A = I$.

Suppose $A = (a_{ik}), a_{ik} \in \{0,1\}, \sum_{k=1}^{K} a_{ik} = 1$ (i.e., A denotes the data membership), then the model reduces to

$$\begin{aligned} O(A,X) &= \| W - AX \|_F^2 \\ &= \text{Trace}[(W-AX)(W-AX)^T] \\ &= \sum_{i=1}^{n}\sum_{j=1}^{m}(w_{ij} - \sum_{k=1}^{K} a_{ik}x_{kj})^2 \\ &= \sum_{i=1}^{n}\sum_{k=1}^{K} a_{ik}\sum_{j=1}^{m}(w_{ij} - x_{kj})^2 \\ &= \sum_{i=1}^{n}\sum_{k=1}^{K} a_{ik}\sum_{j=1}^{m}(w_{ij} - y_{kj})^2 + \sum_{k=1}^{K} p_k \sum_{j=1}^{m}(y_{kj} - x_{kj})^2 \quad (12) \\ &\quad \text{where } p_k = \sum_{i=1}^{n} a_{ik} \text{ and } y_{kj} = \frac{1}{p_k}\sum_{i=1}^{n} a_{ik}w_{ij} \end{aligned}$$

Given A, the objective criterion O is minimized by setting $x_{kj} = y_{kj} = \frac{1}{p_k}\sum_{i=1}^{n} a_{ik}w_{ij}$. Without loss of generality, we assume that the rows belong to a particular cluster are contiguous, so that all data points belonging to the first cluster appear first and the second cluster next, etc [3]. Then A can be represented as $A = \begin{bmatrix} 1 & 0 & \cdots & 0 \\ 1 & 0 & \cdots & 0 \\ \vdots & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \cdots & \vdots \\ 0 & 0 & \cdots & 1 \\ \vdots & \vdots & \cdots & \vdots \\ 0 & 0 & \cdots & 1 \end{bmatrix}$.

Note that $A^TA = \begin{bmatrix} p_1 & 0 & \cdots & 0 \\ 0 & p_2 & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & \cdots & p_K \end{bmatrix}$ is a diagonal matrix with the cluster size on the diagonal. The inverse of A^TA serves as a weight matrix to compute the centroids. Thus we have the following equation for representing centroids

$$X = (A^TA)^{-1}A^TW. \quad (13)$$

On the other hand, given X, $O(A,X)$ is minimized by

$$\hat{a}_{ik} = \begin{cases} 1 & \text{if } \sum_{j=1}^{m}(w_{ij}-y_{kj})^2 < \sum_{j=1}^{m}(w_{ij}-y_{lj})^2 \\ & \text{for } l = 1,\cdots,K, l \neq k \\ 0 & \text{Otherwise} \end{cases} \quad (14)$$

The alternative minimization leads to traditional the K-means clustering procedure [19].

In fact, there are some other variations that can be derived for the one-side clustering. For example, if we prefer a low-dimensional representation of the cluster structure by restricting $Rank(X) = t, t <= min(K-1, m)$, the general model can lead to the one-Side low dimensional clustering [35]. We can also put non-negative constraints on both A and X for other variations [38, 32, 11].

[3] This can be achieved by multiplying W with a permutation matrix if necessary.

4.1.2 Iterative Feature and Data Clustering

When X is identity matrix and if we allow entries of A and B to be any positive values, this leads to the cluster model described in [23]. The objective function can be rewritten as

$$\begin{aligned} O(A,X,B) &= \| W - AB^T \|_F^2 = \text{Trace}((W-AB^T)(W-AB^T)^T) \\ &= \text{Trace}(WW^T) - 2\text{Trace}(WAB^T) + \text{Trace}(AB^TAB^T)) \end{aligned}$$

Note that if we relax A and B and let them be arbitrary matrices, then based on

$$\frac{\partial O}{\partial A} = -WB + AB^TB \quad (15)$$

$$\frac{\partial O}{\partial B} = -W^TA + BA^TA \quad (16)$$

we would get the optimization rules $A = WB(B^TB)^{-1}$ and $B = W^TA(A^TA)^{-1}$. By imposing orthogonal requirements, we could obtain two simplified updating rules which has a natural interpretation analogous to the HITS ranking algorithm [21].

$$B = W^TA, \text{ and } A = WB.$$

Basically, the optimizing rules show a mutually reinforcing relationship between the data and the features for binary dataset which can be naturally expressed as follows: if a feature f (or, data point d) is shared by many points (or, features) that have high weights associated with a cluster c, then feature f (or, data point d) has a high weight associated with c. The clustering approach also share some commonalities with non-negative matrix factorization [22] and concept factorization [11, 37].

4.1.3 Spectral Relaxation

In general, if A and B denote the cluster membership, then $A^TA = diag(p_1,\cdots,p_K)$ and $B^TB = diag(q_1,\cdots,q_C)$ are two diagonal matrices. If we relax the conditions on A and B, requiring $A^TA = I_K$ and $B^TB = I_C$, we would obtain a new variation of two-side clustering algorithm. Note that

$$\begin{aligned} O(A,X,B) &= \| W - AXB^T \|_F^2 = \text{Trace}((W-AXB^T)(W-AXB^T)^T) \\ &= \text{Trace}(WW^T) + \text{Trace}(XX^T) - 2\text{Trace}(AXB^TW^T) \end{aligned}$$

Since $\text{Trace}(WW^T)$ is constant, hence minimizing $O(A,X,B)$ is equivalent to minimizing

$$O'(A,X,B) = \text{Trace}(XX^T) - 2\text{Trace}(AXB^TW^T). \quad (17)$$

The minimum of Equation 17 is achieved where $X = A^TWB$ as $\frac{\partial O'}{\partial X} = X - A^TWB$.

Plugging $X = A^TWB$ into Equation 17, we have

$$\begin{aligned} O'(A,X,B) &= \text{Trace}(XX^T) - 2\text{Trace}(AXB^TW^T) \\ &= \text{Trace}(A^TWBB^TW^TA) - 2\text{Trace}(AA^TWBB^TW^T) \\ &= \text{Trace}(WW^T) - 2\text{Trace}(A^TWBB^TW^TA) \end{aligned}$$

Since the first term $\text{Trace}(WW^T)$ is constant, minimizing $O'(A,X,B)$ is thus equivalent to maximizing $Trace(A^TWBB^TW^TA)$. If we ignore the special structure of A, B and let them be arbitrary orthonormal matrices, the clustering problem then reduced to the trace maximization problem which can be solved by eigenvalue decomposition [39].

A summary of different variations of the general model is listed in Table 2.

Methods	B	A	X	Optimization Procedure
The General Model	$b_{jc} \in \{0,1\}$ $\sum_{j=c}^{C} b_{jc} = 1$	$a_{ik} \in \{0,1\}, \sum_{i=k}^{K} a_{ik} = 1$	$x_{kc} = \frac{\sum_{i=1}^{n}\sum_{j=1}^{m} a_{ik} b_{jc} w_{ij}}{\sum_{i \in n}\sum_{j \in m} a_{ik} b_{jc}}$	Algorithm 1
Block Diagonal Clustering	$b_{jk} \in \{0,1\}$ $\sum_{j=k}^{K} b_{jc} = 1$	$a_{ik} \in \{0,1\}$ $\sum_{i=k}^{K} a_{ik} = 1$	$X = I_{K \times K}$	Algorithm 2
One-Side K-Means	$B = I$	$a_{ik} \in \{0,1\}, \sum_{i=k}^{K} a_{ik} = 1$	$X = (A^T A)^{-1} A^T W$	Alternating Least Square
Iterative Feature Data Clustering	Arbitrary	Arbitrary	$X = I_{K \times K}$	Mutually Reinforcing Optimization
Spectral Relaxation	Orthonormal	Orthonormal	$X = A^T W B$	Two-Side Trace Maximization

Table 2: Summary on Different Variations of the General Model for Binary Data clustering. Each row lists a variation and its associated constraints.

4.2 Relations with Other Clustering Models

In this section, we examine the relations among between our clustering model with other binary clustering models.

4.2.1 Information-Theoretic Clustering

Recently, an information-theoretic clustering framework applicable to empirical joint probability distributions was developed for two-dimensional contingency table or co-occurrence matrix [10]. In this framework, the (scaled) data matrix W is viewed as a joint probability distribution between row and column random variables taking values over the rows and columns. The clustering objective is to seek a hard-clustering of both dimensions such that loss in *mutual information* $I(W) - I(\bar{W})$, where $\bar{W}$ denotes the reduced data matrix, is minimized [34].

Here we explore the relations between our general clustering model and the information-theoretic framework. If we view entries of W as values of a joint probability distribution between row and column random variables, then $I(W) = \sum_{i=1}^{n}\sum_{j=1}^{m} w_{ij} log\frac{w_{ij}}{w_{i.}w_{.j}}$ where $w_{i.} = \sum_{j=1}^{m} w_{ij}$ and $w_{.j} = \sum_{i=1}^{n} w_{ij}$.

Once we have a simplified $K \times C$ matrix $\bar{W}$, we can construct an $n \times m$ matrix $\hat{W}$ as the approximation of original matrix W by

$$\hat{w}_{ij} = \bar{w}_{kc}(\frac{w_{i.}}{\bar{w}_{k.}})(\frac{w_{.j}}{\bar{w}_{.c}}) \tag{18}$$

where $i \in P_k, j \in Q_c$ and $\bar{w}_{k.} = \sum_{c=1}^{C} \bar{w}_{kc}$ and $\bar{w}_{.c} = \sum_{k=1}^{K} \bar{w}_{kc}$. As the approximation preserves marginal probability [10], it can easily check that

$$\bar{w}_{kc} = \sum_{i \in P_k}\sum_{j \in Q_c} \hat{w}_{ij} = \sum_{i \in P_k}\sum_{j \in Q_c} w_{ij} \tag{19}$$

$$\hat{w}_{i.} = w_{i.} \tag{20}$$

$$\hat{w}_{.j} = w_{.j} \tag{21}$$

Hence we have

$$\begin{aligned}
I(\hat{W}_{ij}) &= \sum_{i=1}^{n}\sum_{j=1}^{m} \hat{w}_{ij} log\frac{\hat{w}_{ij}}{w_{i.}w_{.j}} \text{ (Based on Equations 20 and 21)} \\
&= \sum_{i=1}^{n}\sum_{j=1}^{m} \hat{w}_{ij} log\frac{\hat{w}_{ij}\frac{\bar{w}_{kc}}{w_{kc}}}{\bar{w}_{k.}(\frac{w_{i.}}{w_{k.}})\bar{w}_{.c}(\frac{w_{.j}}{w_{.c}})} \\
&= \sum_{i=1}^{n}\sum_{j=1}^{m} \hat{w}_{ij} log\frac{\bar{w}_{kc}}{\bar{w}_{k.}\bar{w}_{.c}} \text{ (Based on Equation 18)} \\
&= \sum_{i=1}^{n}\sum_{j=1}^{m} w_{ij} log\frac{\bar{w}_{kc}}{\bar{w}_{k.}\bar{w}_{.c}} \text{ (Based on Equation 19)} \qquad (22) \\
&= \sum_{k=1}^{K}\sum_{c=1}^{C} \bar{w}_{kc} log\frac{\bar{w}_{kc}}{\bar{w}_{k.}\bar{w}_{.c}} \text{ (Based on Equation 19)} \qquad (23) \\
&= I(\bar{W}) \qquad (24)
\end{aligned}$$

So

$$\begin{aligned}
I(W) - I(\bar{W}) &= I(W) - I(\hat{W}) \text{ (Based on Equation 24)} \\
&= \sum_{i=1}^{n}\sum_{j=1}^{m} w_{ij} log\frac{w_{ij}}{w_{i.}w_{.j}} - \sum_{i=1}^{n}\sum_{j=1}^{m} w_{ij}\frac{\bar{w}_{kc}}{\bar{w}_{k.}\bar{w}_{.c}} \\
&\quad \text{(Based on Equation 22)} \\
&= \sum_{i=1}^{n}\sum_{j=1}^{m} w_{ij} log\frac{w_{ij}}{w_{i.}w_{.j}} - \sum_{i=1}^{n}\sum_{j=1}^{m} w_{ij}\frac{\hat{w}_{ij}}{w_{i.}w_{.j}} \\
&\quad \text{(Based on Equation 18)} \\
&= \sum_{i=1}^{n}\sum_{j=1}^{m} w_{ij} log\frac{w_{ij}}{\hat{w}_{ij}} \\
&\approx \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{m} \frac{(w_{ij} - \hat{w}_{ij})^2}{w_{ij}} \qquad (25)
\end{aligned}$$

The last step from the above derivation is based on power series approximation of logarithm. The approximation is valid if the absolute difference $|w_{ij} - \hat{w}_{ij}|$ are not large as compared with w_{ij}. The right side of Equation 25 can be thought as a weighted version of the right side of Equation 2. Thus minimizing the criterion $O(A,X,B)$ is conceptually consistent with the loss of *mutual information*, i.e., $I(W) - I(\bar{W})$.

4.2.2 Binary Dissimilarity Coefficients

In this section, we show the relations between the block diagonal model with the binary dissimilarity coefficients [4]. A popular partition-based criterion (within-cluster) for one-side clustering is to minimize the summation of distances/dissimilarities inside the cluster. The within-cluster criterion can be described as minimizing

$$S(C) = \sum_{k=1}^{K} \frac{1}{n_k} \sum_{i,i' \in C_k} \delta(w_i, w_{i'}), \tag{26}$$

or [5]

$$S(C) = \sum_{k=1}^{K} \sum_{i,i' \in C_k} \delta(w_i, w_{i'}), \tag{27}$$

where $\delta(w_i, w_{i'})$ is the distance measure between w_i and $w_{i'}$. We use w_i as a point variable and we write $i \in P_k$ to mean that the i-th vector belongs to the k-th data class. For binary clustering, the dissimilarity coefficients are popular measures of the distances.

Various Coefficients: Given two binary data points, w and w', there are four fundamental quantities that can be used to define similarity between the two [4]: $a = \|\{j \mid w_j = w'_j = 1\}\|$, $b = \|\{j \mid$

[4] More discussions on binary dissimilarity coefficients can be found in [24].

[5] Equation 26 computes the weighted sum using the cluster sizes.

$w_j = 1 \wedge w'_j = 0\}\|$, $c = \|\{j \mid w_j = 0 \wedge w'_j = 1\}\|$, and $d = \|\{j \mid w_j = w'_j = 0\}\|$, where $1 \leq j \leq r$. It has been shown in [4] that the presence/absence based dissimilarity measure can be generally written as $D(a,b,c,d) = \frac{b+c}{\alpha a+b+c+\beta d}$, where $\alpha > 0$ and $\beta \geq 0$. Table 3 shows several common dissimilarity coefficients and the corresponding similarity coefficients.

Name	Similarity	Dissimilarity	Metric
Simple Matching Coeff.	$\frac{a+d}{a+b+c+d}$	$\frac{b+c}{a+b+c+d}$	Y
Jaccard's Coeff.	$\frac{a}{a+b+c}$	$\frac{b+c}{a+b+c}$	Y
Dice's Coeff.	$\frac{2a}{2a+b+c}$	$\frac{b+c}{2a+b+c}$	N
Russel&Rao's Coeff.	$\frac{a}{a+b+c+d}$	$\frac{b+c+d}{a+b+c+d}$	Y
Rogers&Tanimoto's Coeff.	$\frac{\frac{1}{2}(a+d)}{\frac{1}{2}(a+d)+b+c}$	$\frac{b+c}{\frac{1}{2}(a+d)+b+c}$	Y
Sokal&Sneath's Coeff. I	$\frac{\frac{1}{2}a}{\frac{1}{2}a+b+c}$	$\frac{b+c}{\frac{1}{2}a+b+c}$	Y
Sokal&Sneath's Coeff. II	$\frac{2(a+d)}{2(a+d)+b+c}$	$\frac{b+c}{2(a+d)+b+c}$	N

Table 3: Binary dissimilarity and similarity coefficients. The "Metric" column indicates whether the given dissimilarity coefficient is metric or not. A 'Y' stands for 'YES' while an 'N' stands for 'No'.

In cluster applications, the rankings based on a dissimilarity coefficient is often of more interest than the actual value of the dissimilarity coefficient. It has been shown that [4], if the paired absences are ignored in the calculation of dissimilarity values, then there is only one single dissimilarity coefficient modulo the global order equivalence: $\frac{b+c}{a+b+c}$. Thus our following discussion is based on the single dissimilarity coefficient.

Relation With Dissimilarity Coefficients: For block diagonal clustering model, given representation (A,B), basically, A denotes the assignments of data points associated into clusters and B indicates the feature representations of clusters. Observe that

$$\begin{aligned} O(A,B) &= ||W - AB^T||_F^2 = \sqrt{\sum_{i,j}(w_{ij} - (AB^T)_{ij})^2} \\ &= \sqrt{\sum_i \sum_j |w_{ij} - (AB^T)_{ij}|^2} = \sqrt{\sum_k^K \sum_{i \in P_k} |w_{ij} - e_{kj}|^2} \\ &= \sqrt{\sum_k^K \sum_{i \in P_k} d(w_i, e_k)}, \end{aligned} \quad (28)$$

where $e_k = (b_{k1}, \cdots, b_{km}), i = 1, \cdots, K$ is the cluster "representative" of cluster P_i. Thus minimizing Equation 28 is the same as minimizing Equation 27 where the distance is defined as $d(w_i, e_k) = \sum_j |w_{ij} - (e_k)_{ij}|^2 = \sum_j |w_{ij} - (e_k)_{ij}|$ (the last equation holds since w_{ij} and $(e_k)_{ij}$ are all binary.) In fact, given two binary vectors X and Y, $\sum_i |X_i - Y_i|$ calculates their mismatches, which is the numerator of their dissimilarity coefficients.

4.2.3 Minimum Description Length

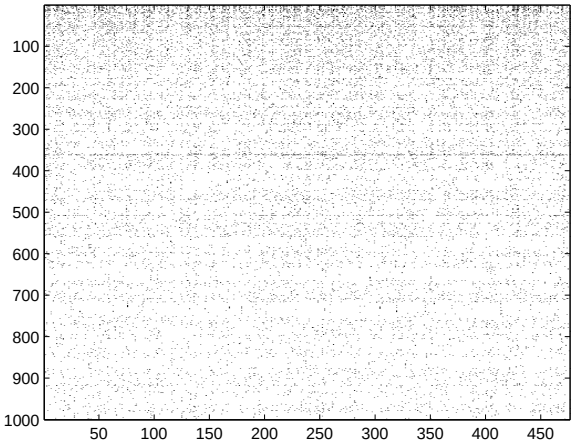
Minimum Description length(MDL) aims at searching for a model that provides the most compact encoding for data transmission [31]. As described in Section 2, in block diagonal clustering, the original matrix W can be approximated by the matrix product of AB^T. Instead of encoding the elements of W alone, we then encode the model, A, B, and the data given the model, $(W|AB^T)$. The overall code length is thus can be expressed as

$$L(W,A,B) = L(W) + L(A) + L(W|AB^T).$$

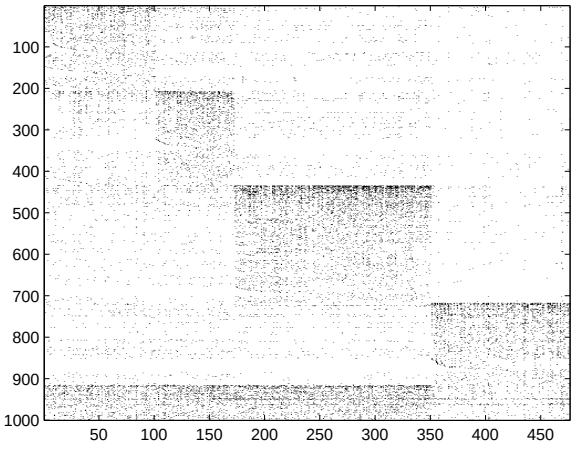
In the Bayesian framework, $L(A)$ and $L(B)$ are negative log priors for A and B and $L(W|AB^T)$ is a negative log likelihood of W given A and B. If we assume that the prior probabilities of all the elements of A and B are uniform (i.e., $\frac{1}{2}$), then $L(A)$ and $L(B)$ are fixed given the dataset W. In other words, we need to use one bit to represent each element of A and B irrespective of the number of 1's and 0's. Hence, minimizing $L(W,A,B)$ reduces to minimizing $L(W|AB^T)$.

PROPOSITION 1. *minimizing $L(W|AB^T)$ is equivalent to minimizing $O(A,B) = \frac{1}{2}||W - AB^T||_F^2$.*

Proposition 1 establishes the connections between MDL and our clustering model. The proof of the proposition is presented in Appendix.



(a) Original Dataset



(b) Dataset after Reordering

Figure 2: Visualization of the original document-data matrix and the reordered document-data matrix. The shaded region represents non-zero entries.

5. DECIDING THE NUMBER OF CLUSTERS

We have seen in the previous section the equivalence among various clustering criteria. In this section, we investigate the problem of determining the number of clusters for binary clustering with our general model. The symmetric association relationship between the data and features provides a novel method for determining the number of clusters for binary data.

5.1 An Example

Given a binary dataset, W, how many possible clusters in the dataset? Let's look at the case where the approximation data take the form of a block diagonal matrix. Based on the symmetric relations between the data and features in binary clustering, the data points in a cluster share many features and vice versa. Hence, if we arrange rows and columns of W based on the cluster assignments (that is, the points and features in the first cluster appear first, the points and features in the second cluster appear next,..., and the points and features in the last cluster appear at the end), then we would get a block diagonal structure.

An example is given in Figure 2 based on CSTR dataset [6], which contains 476 technical report abstracts with 4 different clusters. Each abstract is represented using a 1000-dimension binary vector. For this example, due to large number of terms, it is natural to constrain the features so a term may belong to one cluster only. Figure 2(a) shows the original word-document matrix of CSTR and Figure 2(b) shows the reordered matrix obtained by arranging rows and columns based on the cluster assignments. The four block diagonals in Figure 2(b) correspond to the four clusters and the dense region at the bottom of the figure identifies the feature outliers (which are distributed uniformly across the technical reports. The rough block diagonal sub-structures observed indicate the cluster structure relation between documents and words.

5.2 Number of Clusters

Without loss of generality, we assume that the rows belong to a particular cluster are contiguous, so that all data points belonging to the first cluster appear first and the second cluster next, etc [7]. Similarly, we also ordered the features in W according to which cluster they are in, so that all features belonging to the first cluster appear first and the second cluster next, etc. Hence B has a similar format as A. Note that AB^T is a block diagonal matrix. Assume W has k clusters. Since $W = AB^T + E$, W can be regarded as the addition of two matrices: $W = L + E$ where $L = \begin{pmatrix} W_1 & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 0 & \cdots & 0 & W_k \end{pmatrix} \in R^{n\times m}$, $W_i = \begin{pmatrix} 1 & \cdots & 1 \\ \vdots & \vdots & \vdots \\ 1 & \cdots & 1 \end{pmatrix} \in R^{n_i \times m_i}$ and $E \in R^{n\times m}$ is a matrix with a small value in each entry, i.e., $E = O(\varepsilon)$.

The following theorem gives a way to deciding the number of clusters. The proof of the theorem follows from the standard results in matrix computations (see [14, Theorem 8.3.4. on page 429]).

THEOREM 2. *Let $M = L + E$ where L and E are matrices described above, then M has dominant k singular values.*

Since the permutation does not change the spectral properties, we then could decide the number of clusters based on the singular values of W. In essential, we try to look for a large gap between the singular values σ_k and σ_{k+1} of the related matrices. We note here that the above conclusion can also be derived from matrix perturbation theory [9, 20].

[6]The detailed description of the dataset can be found in Section 6.

[7]This can be achieved by multiplying W with a permutation matrix if necessary.

6. EXPERIMENTS

6.1 Evaluation Measures

There are many ways to measure how accurately clustering algorithm performs. One is the *confusion matrix* [1]. Entry (o,i) of a confusion matrix is the number of data points assigned to output class o and generated from input class i. The *purity* [40] that measures the extend to which each cluster contained data points from primarily one class is also a good metric for cluster quality. The purity of a clustering solution is obtained as a weighted sum of individual cluster purities and is given by $Purity = \sum_{i=1}^{K} \frac{n_i}{n} P(S_i), P(S_i) = \frac{1}{n_i} max_j(n_i^j)$ where S_i is a particular cluster of size n_i, n_i^j is the number of documents of the i-th input class that were assigned to the j-th cluster, K is the number of clusters and n is the total number of points [8]. A high purity value implies that the clusters are "pure" subsets of the input classes. In general, the larger the values of purity, the better the clustering solution is.

6.2 Zoo Dataset

In this section, we evaluate the performance of the general optimization procedure in Section 2.2 on the zoo database available at the UC Irvine Machine Learning Repository. The database contains 100 animals, each of which has 15 boolean attributes and 1 categorical attribute[9]. We translate the numeric attribute, "legs", into six features, which correspond to 0, 2, 4, 5, 6, and 8 legs, respectively. Table 4 shows the confusion matrix of this experiment. In the confusion matrix, we find that the clusters with a large number of animals are likely to be correctly clustered. There are 7 different types in zoo dataset and the animal numbers for each type are 41,20,5,13,3,8,10 respectively. Our procedure, *Algorithm 1*, identifies 5 clusters in the datasets and doesn't identify type 3 (with 5 animals) and type 5 (with 3 animals) due to the limited number of samples. Figure 3 shows the original zoo dataset and the reordered dataset by arranging rows based on their cluster memberships. We can observe the associations between the data and features. For instance, in Figure 3(b), feature 1 is a discriminative feature for cluster 1, feature 8 is a discriminative feature for both cluster 1 and cluster 3 and feature 7 is an outlier feature as it distributes uniformly across all the clusters. Our algorithm explicitly explore the association relationship between data and features and tends to yield better clustering solution. The purity value of our approach, obtained by averaging the results of 10 trials, is 0.94. In comparison, the value is 0.76 for K-means approach.

Output \ Input	1	2	3	4	5	6	7
A	0	20	0	0	0	0	0
B	0	0	0	13	0	0	0
C	41	0	1	0	0	0	0
D	0	0	0	0	0	0	0
E	0	0	0	0	0	0	0
F	0	0	0	0	0	8	2
G	0	0	4	0	3	0	8

Table 4: Confusion matrix of the zoo data.

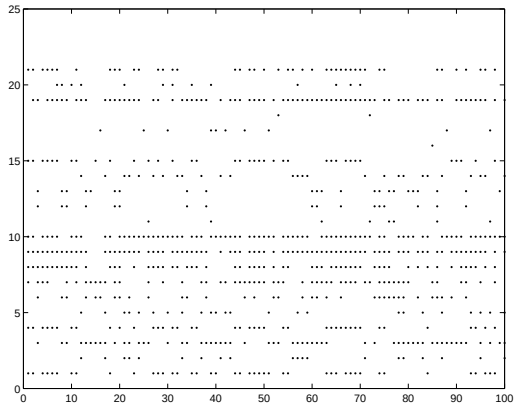
6.3 Clustering Documents

In this section, we apply our clustering algorithm to cluster documents and compare its performance with other standard clustering

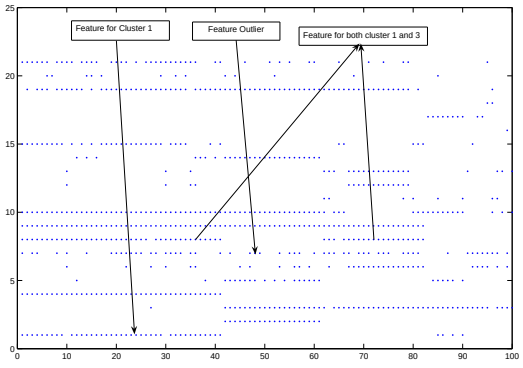
[8]$P(S_i)$ is also called the individual cluster purity.

[9]The original data set has 101 data points but one animal, "frog," appears twice. So we eliminated one of them. We also eliminated two attributes, "animal name" and "type."

algorithms. In our experiments, documents are represented using binary vector-space model where each document is a binary vector in the term space and each element of the vector indicates the presence of the corresponding term. Since there is usually a symmetric association between the documents and words, we use the block diagonal clustering model described in Section 3 in our experiments.



(a) Zoo Dataset



(b) Zoo Dataset after Clustering

Figure 3: Visualization of the zoo dataset and the reordered dataset after clustering. X-axis indicates the animals and Y-axis indicates the features. The shaded region represents non-zero entries.

6.3.1 Document Datasets

We use the following datasets in our experiments and Table 5 summarizes the characteristics of the datasets.

CSTR: This is the dataset of the abstracts of technical reports (TRs) published in the Department of Computer Science at the University of Rochester between 1991 and 2002. The TRs are available at *http://www.cs.rochester.edu/trs*. The dataset contained 476 abstracts, which were divided into four research areas: Natural Language Processing(NLP), Robotics/Vision, Systems, and Theory.

WebKB: The WebKB dataset contains webpages gathered from university computer science departments. There are about 8280 documents and they are divided into 7 categories: student, faculty, staff, course, project, department and other. The raw text is about 27MB. Among these 7 categories, student, faculty, course and project are four most populous entity-representing categories. The associated subset is typically called **WebKB4**. In this paper, we did experiments on both 7-category and 4-category datasets.

Reuters: The Reuters-21578 Text Categorization Test collection contains documents collected from the Reuters newswire in 1987. It is a standard text categorization benchmark and contains 135 categories. In our experiments, we use a subsets of the data collection which include the 10 most frequent categories among the 135 topics and we call it **Reuters-top 10**.

K-dataset: The K-dataset was from WebACE project [16] and it was used in [5] for document clustering. The K-dataset contains 2340 documents consisting news articles from Reuters new service via the Web in October 1997. These documents are divided into 20 classes.

To pre-process the datasets, we remove the stop words use a standard stop list, all HTML tags are skipped and all header fields except subject and organization of the posted article are ignored. In all our experiments, we first select the top 1000 words by mutual information with class labels. The feature selection is done with the rainbow package [28].

Datasets	# documents	# class
CSTR	476	4
WebKB4	4199	4
WebKB	8,280	7
Reuters-top 10	2,900	10
K-dataset	2,340	20

Table 5: Document DataSets Descriptions.

6.3.2 Experimental Results

In our experiments, we compare the performance of our approach on the datasets with the algorithms provided in CLUTO package [40]. Figure 4 shows the performance comparison. Each value is the purity of the corresponding column algorithm on the row dataset. $P1$ is a multi-level partitioning method which tries to maximize the cosine similarity between each document and the cluster centroid. The criterion of $P2$ is similar to minimizing the intra-scatter matrix in discriminant analysis. Hierarchical column shows the results of hierarchical clustering algorithms [10]. We observe that the performance of Algorithm 2 is always either the winner or very close to the winner. The comparison shows that, although there is no single winner on all the datasets, our clustering approach is a viable and competitive algorithm for binary clustering.

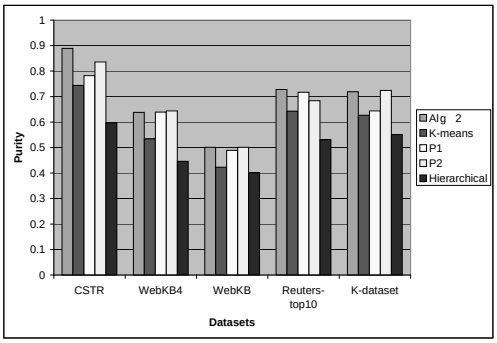


Figure 4: Purity comparisons on various document datasets.

7. RELATED WORK

In this section, we review the clustering methods that are closely related to our approach (see Figure 5 for a summary).

[10]The results reported are the largest values of three different hierarchical clustering algorithms using single-linkage, complete-linkage and UPGMA aggregating policies.

First of all, our clustering model can be regarded as an integration of K-means and boolean factor analysis [29]. It shares the alternating optimization procedure common to K-means type algorithms. The procedure for feature assignment can be thought as an approximation of boolean factor analysis.

Our clustering model is also loosely related to additive clustering where the similarities among the data points are being considered instead of the content of the features [8, 33]. In additive clustering, a similarity model is postulated that expresses the observed similarities of data points in terms of their underlying features. Formally, additive clustering tries to preforms the following matrix decomposition: $S = PWP^T$, where S is the similarity matrix, W is a diagonal matrix indicating the weights of each clusters, and P is a binary matrix indicating the cluster membership of the data points. The matrix P in additive clustering corresponds to the matrix A in our model. However, additive clustering works on similarity matrix instead of the original data matrix and does not admit feature assignments.

Our clustering model performs binary matrix decomposition and can be thought as a special case of positive matrix decomposition [22, 30, 38]. The positive matrix factorization techniques place non-negativity constraints on the data model for optimization. The binary constraints in our model optimization is a special case of non-negativity. The clustering model also shares some common characteristics with probability matrix decomposition models in [26].

Our clustering model is similar to the data and feature maps introduced in [41]. In [41], data and feature maps are two functions from the data and feature sets to the number of clusters and the clustering algorithm is based on *Maximum Likelihood Principle* via co-learning between feature and data maps.

The simultaneous optimization in both directions of data and feature used in our clustering model is similar to the optimization procedure in co-clustering. Govaert [15] studies simultaneous block clustering of the rows and columns of contingency tables. Dhillon et al. [10] propose an information-theoretic co-clustering method for two-dimensional contingency table. The relation between our cluster model with the information-theoretic co-clustering is discussed in section 4.2.1.

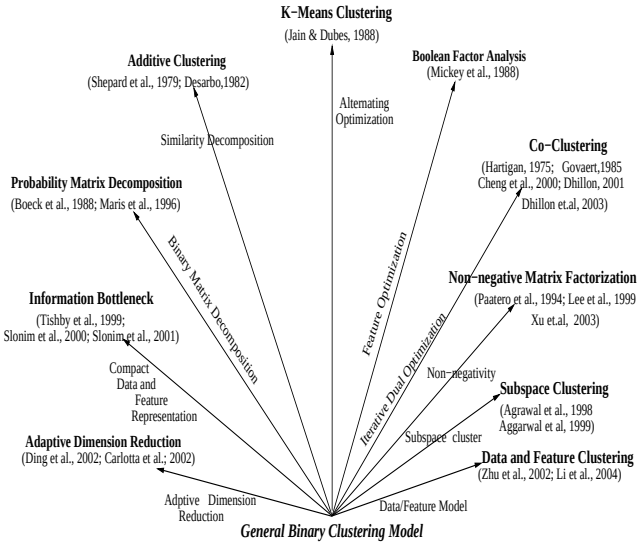


Figure 5: Summary of related work. The arrows show connections.

Since our clustering method explicitly models the cluster structure at each iteration, it is viewed as an adaptive subspace clustering. CLIQUE [2] is an automatic subspace clustering algorithm. It uses equal-size cells and cell density to find dense regions in each subspace in a high dimensional space, where cell size and the density threshold are given as a part of the input. Aggarwal et al. [1] introduce projected clustering and present algorithms for discovering interesting patterns in subspaces of high dimensional spaces. The core idea is a generalization of feature selection which enables selecting different sets of dimensions for different subsets of the data sets. Our clustering method adaptively computes the distance measures and the number of dimensions for each class. It also does not require all projected classes to have the same number of dimensions.

By iteratively updating, our clustering method performs an implicit adaptive feature selection at each iteration and has some common ideas with adaptive feature selection methods. Ding et al. [12] propose an adaptive dimension reduction clustering algorithm. The basic idea is to adaptively update the initial feature selection based on intermediate results during the clustering process and the process is repeated until the best results are obtained. Domeniconi et al. [13] use a Chi-squared distance analysis to compute a flexible metric for producing neighborhoods that are highly adaptive to query locations. Neighborhoods are elongated along less relevant feature dimensions and constricted along most influential ones.

As discussed in Section 4.2.1, our clustering model can be thought of as an approximate iterative information bottleneck method. The information bottleneck (IB) framework is first introduced for one-sided clustering [36]. The core idea of IB is as follows: given the empirical joint distribution of two variables (X,Y), one variable is compressed so that the mutual information about the other is preserved as much as possible. The IB algorithm in [36] tries to minimize the quantity $I(X;\hat{X})$ while maximizing $I(\hat{X};Y)$, where I is the mutual information and $\hat{X}$ is the cluster representation of X.

8. CONCLUSION

In this paper, we introduce a general binary clustering model that allows explicit modeling of the feature structure associated with each cluster. An alternating optimization procedure is employed to perform two tasks: optimization of the cluster structure and updating of the clusters. We provide several variants of the general clustering model using different characterizations. We also provide a unified view on binary clustering by establishing the connections among various clustering approaches. Experimental results on document datasets suggest the effectiveness of our approach.

Acknowledgment

The author is grateful to Dr. Shenghuo Zhu for his insightful suggestions. The author also wants to thank the anonymous reviewers for their invaluable comments.

9. REFERENCES

[1] Aggarwal, C. C., Wolf, J. L., Yu, P. S., Procopiuc, C., & Park, J. S. (1999). Fast algorithms for projected clustering. *Proceedings of the 1999 ACM SIGMOD international conference on Management of data (SIGMOD'99)* (pp. 61–72). ACM Press.

[2] Agrawal, R., Gehrke, J., Gunopulos, D., & Raghavan, P. (1998). Automatic subspace clustering of high dimensional data for data mining applications. *Proceedings of the 1998 ACM SIGMOD international conference on Management of data (SIGMOD'98)* (pp. 94–105). ACM Press.

[3] Baier, D., Gaul, W., & Schader, M. (1997). Two-mode overlapping clustering with applications to simultaneous benefit segmentation and market structuring. In R. Klar and O. Opitz (Eds.), *Classification and knowledge organization*, 577–566. Springer.

[4] Baulieu, F. B. (1997). Two variant axiom systems for presence/absence based dissimilarity coefficients. *Journal of Classification, 14*, 159–170.

[5] Boley, D., Gini, M., Gross, R., Han, E.-H., Hastings, K., Karypis, G., Kumar, V., Mobasher, B., & Moore, J. (1999). Document categorization and query generation on the world wide web using webace. *AI Review, 13*, 365–391.

[6] Castillo, W., & Trejos, J. (2002). Two-mode partitioning: Review of methods and application and tabu search. In K. Jajuga, A. Sokolowski and H.-H. Bock (Eds.), *Classification, clustering and data analysis*, 43–51. Springer.

[7] Cho, H., Dhillon, I. S., Guan, Y., & Sra, S. (2004). Minimum sum-squared residue co-clustering of gene experssion data. *Proceedings of the SIAM Data Mining Conference*.

[8] Desarbo, W. (1982). GENNCLUS: New models for general nonhierarchical clustering analysis. *Psuchometrika, 47*, 449–475.

[9] Deuflhard, P., Huisinga, W., Fischer, A., & Schutte, C. (2000). Identification of almost invariant aggregates in reversible nearly coupled markov chain. *Linear Algebra and Its Applications, 315*, 39–59.

[10] Dhillon, I. S., Mallela, S., & Modha, S. S. (2003). Information-theoretic co-clustering. *Proceedings of the 9th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (SIGKDD 2003)* (pp. 89–98). ACM Press.

[11] Dhillon, I. S., & Modha, D. S. (2001). Concept decompositions for large sparse text data using clustering. *Machine Learning, 42*, 143–175.

[12] Ding, C., He, X., Zha, H., & Simon, H. (2002). Adaptive dimension reduction for clustering high dimensional data. *Proceedings of the 2002 IEEE International Conference on Data Mining (ICDM 2002)* (pp. 107–114). IEEE Computer Society.

[13] Domeniconi, C., Peng, J., & Gunopulos, D. (2002). Locally adaptive metric nearest-neighbor classification. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 24*, 1281–1285.

[14] Golub, G. H., & Loan, C. F. V. (1991). *Matrix computation*s. The Johns Hopkins University Press.

[15] Govaert, G. (1995). Simultaneous clustering of rows and columns. *Control and Cybernetics, 24*, 437–458.

[16] Han, E.-H., Boley, D., Gini, M., Gross, R., Hastings, K., Karypis, G., Kumar, V., Mobasher, B., & Moore, J. (1998). WebACE: A web agent for document categorization and exploration. *Proceedings of the 2nd International Conference on Autonomous Agents (Agents'98)*. ACM Press.

[17] Hartigan, J. A. (1975). *Clustering algorithms*. Wiley.

[18] Hastie, T., Tibshirani, R., & Friedman, J. (2001). *The elements of statistical learning: Data mining, inference, prediction.* Springer.

[19] Jain, A. K., & Dubes, R. C. (1988). *Algorithms for clustering data*. Prentice Hall.

[20] Kato, T. (1995). *Perturbation theory for linear operators*. Springer.

[21] Kleinberg, J. M. (1999). Authoritative sources in a hyperlinked environment. *Journal of the ACM, 46*, 604–632.

[22] Lee, D. D., & Seung, H. S. (2000). Algorithms for non-negative matrix factorization. *NIPS* (pp. 556–562).

[23] Li, T., & Ma, S. (2004). IFD:iterative feature and data clustering. *Proceedings of the 2004 SIAM International conference on Data Mining (SDM 2004)*. SIAM.

[24] Li, T., Ma, S., & Ogihara, M. (2004b). Entropy-based criterion in categorical clustering. *Proceedings of The 2004 IEEE International Conference on Machine Learning (ICML 2004)*. 536-543.

[25] Li, T., & Zhu, S. (2005). On clustering binary data. *Proceedings of the 2005 SIAM International Conference On Data Mining(SDM'05)* (pp. 526–530).

[26] Maris, E., Boeck, P. D., & Mechelen, I. V. (1996). Probability matrix decomposition models. *Psychometrika, 61*, 7–29.

[27] Maurizio, V. (2001). Double k-means clustering for simultaneous classification of objects and variables. In S. Borra, R. Rocci, M. Vichi and M. Schader (Eds.), *Advances in classification and data analysis*, 43–52. Springer.

[28] McCallum, A. K. (1996). Bow: A toolkit for statistical language modeling, text retrieval, classification and clustering. http://www.cs.cmu.edu/ mccallum/bow.

[29] Mickey, M. R., Mundle, P., & Engelman, L. (1988). Boolean factor analysis. In *Bmdp statistical software manual*, vol. 2, 789–800. University of California Press.

[30] Paatero, P., & Tapper, U. (1994). Positive matrix factorization: A non-negative factor model with optimal utilization of error estimates of data values. *Environmetrics, 5*, 111–126.

[31] Rissanen, J. (1978). Modeling by shortest data description. *Automatica, 14*, 465–471.

[32] Sha, F., Saul, L. K., & Lee, D. D. (2002). Multiplicative updates for nonegative quadratic programming in support vector machines. *Advances in Neural Information Processing Systems* (pp. 1065–1072).

[33] Shepard, R. N., & Arabie, P. (1979). Additive clustering: Representation of similarities as combinations of discrete overlapping properties. *Psychological Review, 86*, 87–123.

[34] Slonim, N., & Tishby, N. (2000). Document clustering using word clusters via the information bottleneck method. *Proceedings of the 23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval(SIGIR'00)* (pp. 208–215). ACM Press.

[35] Soete, G., & Carroll, J. D. (1994). K-means clustering in a low-dimensional eucildean space. *New Approaches in Classification and Data Analysis* (pp. 212–219). Springer-Verlag.

[36] Tishby, N., Pereira, F. C., & Bialek, W. (1999). The information bottleneck method. *Proc. of the 37-th Annual Allerton Conference on Communication, Control and Computing* (pp. 368–377).

[37] Xu, W., & Gong, Y. (2004). Document clustering by concept factorization. *SIGIR '04: Proceedings of the 27th annual international conference on Research and development in information retrieval* (pp. 202–209). Sheffield, United Kingdom: ACM Press.

[38] Xu, W., Liu, X., & Gong, Y. (2003). Document clustering based on non-negative matrix factorization. *Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval(SIGIR'03)* (pp. 267–273). ACM Press.

[39] Zha, H., He, X., Ding, C., & Simon, H. (2001). Spectral relaxation for k-means clustering. *Proceedings of Neural Information Processing Systems*.

[40] Zhao, Y., & Karypis, G. (2002). *Evaluation of hierarchical clustering algorithms for document datasets* (Technical Report). Department of Computer Science, University of Minnesota.

[41] Zhu, S., Li, T., & Ogihara, M. (2002). CoFD: An algorithm for non-distance based clustering in high dimensional spaces. *Proceedings of the Fourth International Conference on Data Warehousing and Knowledge Discovery (DaWak2002)* (pp. 52–62).

Appendix: Proof of Proposition 1

Proof: Use $\hat{W}$ to denote the generated data matrix by A and B. For all i, $1 \le i \le n$, j, $1 \le j \le m$, $b \in \{0,1\}$, and $c \in \{0,1\}$, we consider $p(x_{ij} = b \mid \hat{w}_{ij}(A,B) = c)$, the probability of the original data $W_{ij} = b$ conditioned upon the generated data $(\hat{w})_{ij}$, via AB^T, is c. Note that

$$p(w_{ij} = b \mid \hat{w}_{ij}(A,B) = c) = \frac{N_{bc}}{N_{\cdot c}}.$$

Here N_{bc} is the number of elements of W which have value b where the corresponding value for $\hat{W}$ is c, and $N_{\cdot c}$ is the number of elements of $\hat{W}$ which have value c. Then the code length for $L(W,A,B)$ is

$$\begin{aligned} L(W,A,B) &= -\sum_{b,c} N_{bc} \log P(w_{ij} = b \mid \hat{w}_{ij}(A,B) = c) \\ &= -nm \sum_{b,c} \frac{N_{bc}}{nm} \log \frac{N_{bc}}{N_{\cdot c}} = nmH(W|\hat{W}(A,B)) \end{aligned}$$

So minimizing the coding length is equivalent to minimizing the conditional entropy. Denote $p_{bc} = p(w_{ij} = b \mid \hat{w}_{ij}(A,B) = c)$. We wish to find the probability vectors $\mathbf{p} = (p_{00}, p_{01}, p_{10}, p_{11})$ that minimize

$$H(W|\hat{W}(A,B)) = -\sum_{i,j \in \{0,1\}} p_{ij} \log p_{ij} \tag{29}$$

Since $-p_{ij} \log p_{ij} \ge 0$, with the equality holding at $p_{ij} = 0$ or 1, the only possible probability vectors which minimize $H(W|\hat{W}(D,F))$ are those with $p_{ij} = 1$ for some i, j and $p_{i_1 j_1} = 0, (i_1, j_1) \ne (i, j)$. Since $\hat{W}$ is an approximation of W, it is natural to require that p_{00} and p_{11} be close to 1 and p_{01} and p_{10} be close to 0. This is equivalent to minimizing the mismatches between W and $\hat{W}$, i.e., minimizing $O(A,B) = ||W - AB^T||_F^2$. □