The Multi-Feature Information Bottleneck with Application to Unsupervised Image Categorization

Zhengzheng Lou, Yangdong Ye, Xiaoqiang Yan
School of Information Engineering, Zhengzhou University, China
zzlou@zzu.edu.cn, yeyd@zzu.edu.cn, iexqyan@gmail.com

Abstract

We present a novel unsupervised data analysis method, *Multi-feature Information Bottleneck (MfIB)*, which is an extension of the *Information Bottleneck (IB)*. In comparison with the original IB, the proposed MfIB method can analyze the data simultaneously from multiple feature variables, which characterize the data from multiple cues. To verify the effectiveness of MfIB, we apply the corresponding MfIB algorithm to unsupervised image categorization. In our experiments, by taking into account multiple types of features, such as local shape, color and texture, the MfIB algorithm is found to be consistently superior to the original IB algorithm which takes only one source of features into consideration. Besides, the performance of MfIB algorithm is also superior to the state-of-the-art unsupervised image categorization methods.

1 Introduction

The *Information Bottleneck* (IB) method [Tishby *et al.*, 1999] is one of the popular and powerful unsupervised data analysis techniques. In the IB framework, the data and its features are treated as the instances of two random variables X and Y, of which the joint distribution $p(X, Y)$ can be empirically estimated from the "co-occurrence" or "dyadic" data matrix. Then the data analysis problem is viewed as the process of compressing X variable into a "bottleneck" variable T and the compressing results $p(t|x)$ finally reflect the hidden patterns of the data. The IB algorithms have been applied in many fields and the results have demonstrated that the IB method is a promising technique for discovering the underlying patterns resided in the data set [Slonim *et al.*, 2002; Slonim, 2002; Lou *et al.*, 2010].

While compressing X to T, the IB method tries to preserve the information about the relevant variable Y as much as possible. The variable Y specifies what information the compressed representation T should preserve. Consider a simple example where X and Y denote the documents and words respectively. If our task is to compress document variable X, in the IB framework, the compression variable T should preserve the information about word variable Y as high as possible. Since the words carry the semantic information, the final compressing results $p(t|x)$ will reveal the topic patterns of the documents. If a new feature variable Y is available, the IB method will obtain a different compressed representation according to the information provided by Y.

The feature variable Y is of great significance to the IB method. One variable Y, which has the power to perfectly discriminate the classes in the data, will yield a good compressing results for IB algorithms. Therefore, to improve the performance of IB algorithms, one alternative way is to seek a better data representation. There are many works that aim to learn more discriminative features, such as feature weighting [Salton, 1991], feature clustering [Slonim and Tishby, 2000] and feature selection [Dhillon *et al.*, 2004]. We can use some of them to learn a more discriminative feature variable for IB algorithms. However, all these works only consider one type of features. In real applications, there may be several cues to complementally characterize the same object. For example, we can use both words and photos to describe one person. The words can describe the person from the information of age, sex, height and other characters, while the photos can show the appearance of one person directly. Words and photos are two different feature types. Even though we can make use of the words to describe the appearance of one person, and guess the vague age of the person from the photo, such information is not perfectly accurate and direct. So, we would rather adopt both words and photos than adopt only one source of them to describe people complementally. In the IB framework, there is only one feature variable Y to denote one type of features, and the data analysis task can only be performed on one feature variable. While only one feature type can't completely characterize the data, the relevant information provided to IB is limited and the final compressing results will be poor to reveal the hidden patterns. So, can we analyze the data from multiple feature types simultaneously?

In this paper, we extend the original IB method to the *Multi-feature Information Bottleneck* (MfIB), which can simultaneously process multiple feature types and analyze the data from multiple cues. In the MfIB framework, each type of features is denoted by one feature variable and there is a corresponding joint distribution between the data variable and feature variable. Instead of only maximally preserving the information of one feature variable, the MfIB tries to simultaneously maintain the information of multiple feature variables while X is compressed to T. Therefore, the compressing results can simultaneously reflect the hidden patterns provided by multiple cues of features, and the multiple complemental variables can help the IB method to extract the patterns of the data that are much closer to the real patterns resided in the data.

In order to verify the effectiveness of the proposed MfIB method, we apply the MfIB algorithm to the field of unsupervised image categorization, of which the task is to discover object or scene categories from a collection of unlabeled images without any supervision [Sivic *et al.*, 2005; Lou *et al.*, 2010; Tuytelaars *et al.*, 2010]. The first step, which is also one of key issues to understand the semantics of the images, is to select one feature extraction technique. In the computer vision field, there are many feature extraction techniques, such as SIFT [Lowe, 2004], SURF [Bay *et al.*, 2008], Color Attention [Khan *et al.*, 2009], TPLBP [Wolf *et al.*, 2008] and so on. Each technique can extract some information from one aspect of the images. For example, SIFT and SURF extract the local shape information, while Color Attention and TPLBP extract the color and texture information respectively. Even though both SIFT and SURF extract the shape information, they don't have the same power to discriminate the image categories because of the differences between the corresponding feature extraction algorithms. Among these feature types, we can't determine which one is better than the others and their powers to discriminate the classes are also different. So, if we categorize the images on multiple complemental feature types, the performance may be improved. [Lou *et al.*, 2010] have demonstrated that the IB method is a powerful technique for unsupervised image categorization. In this paper, we extend the IB to MfIB and apply the proposed MfIB algorithm to unsupervised image categorization. The MfIB algorithm can simultaneously process multiple feature variables, which correspond to multiple feature types extracted from the images. The experiments on 7 benchmark image data sets show that, by combining multiple feature variables, the MfIB clearly outperforms the original IB method. In addition, the performance of MfIB is also superior to the state-of-the-art unsupervised image categorization methods [Sivic *et al.*, 2005; Lou *et al.*, 2010].

The main contributions of this paper can be summarized as follows:

- We extend the original Information Bottleneck method to Multi-feature Information Bottleneck, which can fuse multiple aspects of information from multiple cues into the final data analysis results, and thus captures the complementary information residing in multiple features.
- We apply the proposed MfIB algorithm to unsupervised image categorization, which provides a solution to the problem of unsupervised image categorization by combing diverse multiple feature types.

2 The Information Bottleneck Method

The Information Bottleneck [Tishby *et al.*, 1999] is an information-theoretic based data analysis method, which treats the pattern extraction from data as a process of data compression. Assume that we are given a collection of unlabeled data $\mathcal{X} = \{x_1, x_2, \cdots, x_m\}$ and its co-occurrence features $\mathcal{Y} = \{y_1, y_2, \cdots, y_n\}$, where m and n are the total number of samples and the size of features respectively. Let X and Y be two discrete random variables, taking values from $\mathcal{X}$ and $\mathcal{Y}$, respectively. Then, for every $x \in \mathcal{X}$, we can define the conditional distribution of the features as $p(y|x) = \frac{n(x,y)}{\sum_{y'} n(x,y')}$, where $n(x,y)$ denotes the number of occurrences of feature y in the sample x. If the prior distribution of $p(x)$ are given, we can obtain the joint distribution between X and Y by $p(x,y) = p(y|x)p(x)$.

Based on the above joint distribution, Tishby et al. [Tishby *et al.*, 1999] formulate the data analysis problem as looking for a compressed representation T of X which maintains the information about the relevant variable Y as high as possible. The compactness of the representation and the preservation of the relevant information are measured by the mutual information $I(T;X)$ and $I(T;Y)$, respectively. The mutual information between variables X and Y are defined as [Cover and Thomas, 1991]:

$$I(X;Y) = \sum_{x\in X}\sum_{y\in Y} p(x,y)\log\frac{p(x,y)}{p(x)p(y)}. \tag{1}$$

Formally, Tishby et al. suggest the following IB-functional:

$$\mathcal{L}_{min} = I(T;X) - \beta \cdot I(T;Y), \tag{2}$$

where β is the Lagrange multiplier controlling the trade-off between the compression from X to T and the preserved information of T about Y. The formal solution to the IB-functional (2) is given by the following equations which must be solved self-consistently,

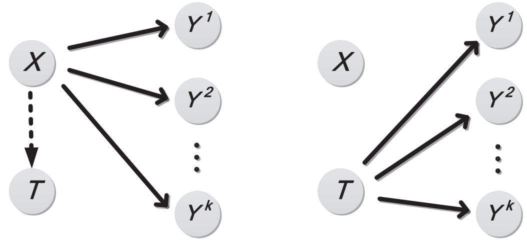
$$\begin{cases} p(t|x) = \frac{p(t)}{Z(x,\beta)} e^{-\beta D_{KL}[p(y|x)||p(y|t)]} \\ p(y|t) = \frac{1}{p(t)}\sum_x p(x,y,t) = \frac{1}{p(t)}\sum_x p(x,y)p(t|x) \\ p(t) = \sum_{x,y} p(x,y,t) = \sum_x p(x)p(t|x), \end{cases} \tag{3}$$

where $D_{KL}[p(y|x)||p(y|t)] = \sum_y p(y|x)\log\frac{p(y|x)}{p(y|t)}$ is the *Kullback-Leibler(KL) divergence* [Cover and Thomas, 1991] between the conditional distributions $p(y|x)$ and $p(y|t)$, $Z(x,\beta)$ is a normalization function. Obviously, the variables $p(t)$ and $p(y|t)$ are determined through $p(t|x)$.

3 The Multi-feature Information Bottleneck

The original IB method processes only one feature variable Y. In this section, we present a new IB framework, Multi-feature Information Bottleneck (MfIB), which can simultaneously process multiple feature variables. For clarity, we first define the task of MfIB.

Definition 1 (MfIB). *Given a discrete random variable X, taking values from $\mathcal{X} = \{x_1, x_2, \cdots, x_m\}$, there are $k(k \geq 1)$ discrete random variables $Y^1, \cdots, Y^k$ and the corresponding joint distributions $p(X,Y^1), \cdots, p(X,Y^k)$ $(1 \leq i \leq k)$. Each variable Y^i takes values from one feature source $\mathcal{Y}^i = \{y_1^i, y_2^i, \cdots, y_{n_i}^i\}$ to characterize the samples of $\mathcal{X}$ from one cue. The task of MfIB is to learn a good compressing representation $p(t|x)$ of X to T from multiple feature variables $Y^1, \cdots, Y^k$.*



(a) Data compressing

(b) Information preservation

Figure 1: The model of MfIB. (a) The data compressing shows the compression relationship among variables. The solid arrow from X to Y^i ($1 \leq i \leq k$) denotes that there is a known joint distribution $p(X, Y^i)$ between the sample variable X and feature variable Y^i. The dotted arrow from X to T denotes that the variable X is compressed to variable T, which is represented by conditional distribution $p(t|x)$. (b) The information preservation specifies what information the compressed variable T should preserve.

3.1 Objective Function for MfIB

The MfIB aims to simultaneously process multiple feature variables. Figure 1 is the model of MfIB. From this figure, we can see that there are multiple feature variables related to the data variable X, and the compressed variable T should simultaneously preserve the information about feature variables $Y^1, \cdots, Y^k$ as much as possible. When $k = 1$, the MfIB degenerates to the original IB method, which indicates that the MfIB is a general framework for multiple feature variables extension of the IB method. The objective function of MfIB is formulated as

$$\begin{aligned}\mathcal{L}_{max}[p(t|x)] &= [\lambda_1 \cdot I(T;Y^1) + \cdots + \lambda_k \cdot I(T;Y^k)] \\ &- \beta^{-1} \cdot I(T;X), \end{aligned} \tag{4}$$

where $I(T;X)$ measures the compactness of the new representation T, $\lambda_1 \cdot I(T;Y^1) + \cdots + \lambda_k \cdot I(T;Y^k)$ measures the preserved relevant information. $\beta \geq 0$ is the balance parameter controlling the trade-off between compression and information preservation. $\lambda_i \geq 0 (1 \leq i \leq k)$ are trade-off parameters to balance the influence among different feature variables.

From the objective function (4), we can see that the multiple feature variables are embedded into the IB framework. Thus, the MfIB can simultaneously process multiple types of features and mine the underlying patterns hidden in the data $\mathcal{X}$ from multiple cues. To analyze the data, the number of clusters M is much less than the original data size $|\mathcal{X}|$ (i.e. $M \ll |\mathcal{X}|$, which implies a significant compression. In this paper, we only concentrate on maximally preserving the relevant feature information $\lambda_1 \cdot I(T;Y^1) + \cdots + \lambda_l \cdot I(T;Y^k)$, and set $\beta = \infty$. Then we rewrite the objective function of MfIB as

$$\mathcal{L}_{max}[p(t|x)] = \lambda_1 \cdot I(T;Y_1) + \cdots + \lambda_l \cdot I(T;Y_l). \tag{5}$$

Our remaining task is to maximize the value of the objective function Equation (5). However, maximizing Equation (5) is not an easy task, since it is non-convex and there are no good solutions currently to directly optimize this objective function. In this work, we only consider the "hard" clustering, where the value of $p(t|x)$ is either 0 or 1. Thus, the task of MfIB becomes to find an optimal partition of $\mathcal{X}$, which should maximally preserve the information in objective function (5). To realize this, we will adopt a sequential "draw-and-merge" optimization procedure [Slonim *et al.*, 2002] to optimize the objective function, which is guaranteed to converge to a local maximum of the information.

3.2 Optimization for MfIB

The sequential "draw-and-merge" procedure starts with a random partition of $\mathcal{X}$ into M clusters. At each step, a single $x \in \mathcal{X}$ is "drawn" from its current cluster t^{old} and is represented as a new single cluster. Now, we have $M + 1$ clusters. To ensure that the number of clusters is M, we must "merge" x into one of the clusters. The goal of the algorithm is to maximize the objective function (5), so each "draw-and-merge" procedure should improve the value of objective function $\mathcal{L}$ in (5). Therefore, we must choose the best cluster t^{new} for x to be merged. In the following, we will give the solution to this problem. First, we have the following proposition.

Proposition 1. *: Let x be merged into cluster t and become a new cluster $\tilde{t}$, i.e. $\{\{x\}, t\} \Rightarrow \tilde{t}$. Then*

$$p(\tilde{t}) = p(x) + p(t), \tag{6}$$

$$p(Y^i|\tilde{t}) = \frac{p(x)}{p(\tilde{t})} p(Y^i|x) + \frac{p(t)}{p(\tilde{t})} p(Y^i|t), \tag{7}$$

where $1 \leq i \leq k$.

The basic question in sequential "draw-and-merge" process is of course which cluster x should be merged into at each step. The value of objective function (5) is changed when x is drawn from its current cluster or merged into one of clusters. Let $\mathcal{L}^{old}$ and $\mathcal{L}^{mid}$ denote the value of objective function (5) before and after the draw step of x. Let $\mathcal{L}^{new}$ denote the value of (5) after x is merged into some cluster t.

Now, we calculate the difference between the values of $\mathcal{L}^{mid}$ and $\mathcal{L}^{new}$, which is also called the merge cost $d_{\mathcal{L}}(\{x\}, t)$ in our work.

$$\begin{aligned} d_{\mathcal{L}}(\{x\}, t) &= \Delta\mathcal{L} = \mathcal{L}^{mid} - \mathcal{L}^{new} \\ &= [\lambda_1 \cdot I(T^{mid};Y^1) + \cdots + \lambda_k \cdot I(T^{mid};Y^k)] - \\ &\quad [\lambda_1 \cdot I(T^{new};Y^1) + \cdots + \lambda_k \cdot I(T^{new};Y^k)] \\ &= \lambda_1 \cdot [I(T^{mid};Y^1) - I(T^{new};Y^1)] + \\ &\quad \cdots + \lambda_k \cdot [I(T^{mid};Y^k) - I(T^{new};Y^k)] \\ &= \lambda_1 \cdot \Delta I^1 + \cdots + \lambda_k \cdot \Delta I^k, \end{aligned}$$

where

$$\begin{aligned} \Delta I^i &= I(T^{mid};Y^i) - I(T^{new};Y^i) \\ &= p(x) \sum_{y^i \in \mathcal{Y}^i} p(y^i|x) \log \frac{p(y^i|x)}{p(y^i)} + p(t) \sum_{y^i \in \mathcal{Y}^i} p(y^i|t) \log \frac{p(y^i|t)}{p(y^i)} \\ &\quad - p(\tilde{t}) \sum_{y^i \in \mathcal{Y}^i} p(y^i|\tilde{t}) \log \frac{p(y^i|\tilde{t})}{p(y^i)}. \end{aligned}$$

Using Proposition 1, we obtain

$$
\begin{aligned}
&\Delta I^i = \\
&p(x) \sum_{y^i \in \mathcal{Y}^i} p(y^i|x) \log \frac{p(y^i|x)}{p(y^i)} + p(t) \sum_{y^i \in \mathcal{Y}^i} p(y^i|t) \log \frac{p(y^i|t)}{p(y^i)} \\
&- \sum_{y^i \in \mathcal{Y}^i} p(x)p(y^i|x) \log \frac{p(y^i|\tilde{t})}{p(y^i)} - \sum_{y^i \in \mathcal{Y}^i} p(t)p(y^i|t) \log \frac{p(y^i|\tilde{t})}{p(y^i)} \\
&= p(x) \sum_{y^i \in \mathcal{Y}^i} p(y^i|x) \log \frac{p(y^i|x)}{p(y^i|\tilde{t})} + p(t) \sum_{y^i \in \mathcal{Y}^i} p(y^i|t) \log \frac{p(y^i|t)}{p(y^i|\tilde{t})} \\
&= p(x) D_{KL}[p(Y^i|x)||p(Y^i|\tilde{t})] + p(t) D_{KL}[p(Y^i|t)||p(Y^i|\tilde{t})] \\
&= [p(x) + p(t)] \cdot JS_{\Pi}[p(Y^i|x), p(Y^i|t)],
\end{aligned}
$$

where

$$
\begin{aligned}
&JS_{\Pi}[p(Y^i|x), p(Y^i|t)] = \\
&\pi_1 D_{KL}[p(Y^i|x)||p(Y^i|t)] + \pi_2 D_{KL}[p(Y^i|t)||p(Y^i|t)]
\end{aligned}
$$

is the *Jensen-Shannon* divergence [Cover and Thomas, 1991], $\Pi = \{\pi_1, \pi_2\} = \{\frac{p(x)}{p(x)+p(t)}, \frac{p(t)}{p(x)+p(t)}\}$.

In this paper, we use JS_i to denote $JS[p(Y^i|x), p(Y^i|t)]$ for simplicity. Similar analysis will yield

$$
\begin{aligned}
d_{\mathcal{L}}(\{x\}, t) &= \lambda_1 \cdot \Delta I^1 + \cdots + \lambda_k \cdot \Delta I^k \\
&= [p(x) + p(t)] \cdot [\lambda_1 \cdot JS_1 + \cdots + \lambda_k \cdot JS_k]. \qquad (8)
\end{aligned}
$$

Because $JS_i \geq 0$ [Cover and Thomas, 1991], $d_{\mathcal{L}}(\{x\}, t) \geq 0$. Therefore, when some x is merged into one of clusters, there must be some information lost. To maximally preserve information, in the merging step, we will choose the cluster that makes the minimal loss of information. That is x will be merged into the cluster t^{new} such that $t^{new} = \arg\min_{t \in \mathcal{T}} d_{\mathcal{L}}(\{x\}, t)$. The details of MfIB algorithm is shown in Algorithm 1.

Algorithm 1 The Multi-feature Information Bottleneck Algorithm: MfIB

```
1:  Input: Joint distributions $p(X, Y^1), \cdots, p(X, Y^k)$,
    trade-off parameters $\lambda^1, \cdots, \lambda^k$, number of clusters $M$.
2:  Output: A partition $T$ of $\mathcal{X}$ into $M$ clusters.
3:  Initialize:
4:  $T \leftarrow$ Random partition of $\mathcal{X}$ into $M$ clusters;
5:  Procedure:
6:  repeat
7:    for For every $x \in \mathcal{X}$ do
8:      Remove $x$ from current cluster $t(x)$;
9:      For data point $x$, calculate merge costs $d_{\mathcal{L}}(\{x\}, t)$ of
        all possible reassignments of $x$ to different clusters
        based on Equation (8);
10:     Merge $x$ into cluster $t^{new}$ such that $t^{new} =$
        $\arg\min_{t \in \mathcal{T}} d_{\mathcal{L}}(\{x\}, t)$;
11:   end for
12: until Convergence
```

In Algorithm 1, each "draw-and-merge" iteration will merge x into the cluster t^{new} such that $t^{new} = \arg\min_{t \in \mathcal{T}} d_{\mathcal{L}}(\{x\}, t)$, so this step will improve the value of object function (5). For each Y_i, $I(T; Y_i) \leq I(X; Y_i)$, so the value of objection function (5) is upper bounded. Therefore, MfIB algorithm will converge in a finite number of iterations. Note that, although MfIB algorithm is able to increase the value of (5), it is only able to converge to a local maximum of the information in Equation (5). Finding the global optimal solution is NP-hard.

Table 1: The image data sets

Data Sets	number of categories	size of data
Soccer	7	280
MSRC	8	240
Sports	8	1576
17flowers	17	1360
Dslr	31	489
Webcam	31	795
Amazon	31	2813

3.3 Complexity Analysis

We now analyze the computational cost of our proposed MfIB algorithm showed in Algorithm 1. At step 9, we should calculate $d_{\mathcal{L}}(\{x\}, t)$ for every t which takes $O(M(|\mathcal{Y}^1| + \cdots + |\mathcal{Y}^k|))$. The time complexity of our algorithm is $O(LM|\mathcal{X}|(|\mathcal{Y}^1| + \cdots + |\mathcal{Y}^k|))$, where L is the number of repetitions that should be performed over X until convergence is attained. In the following experiments, we will show that MfIB algorithm will take a few numbers of repetitions to coverage a local optimal value of objective function (5). Usually, the number of clusters M can be considered as constant. So the time complexity of MfIB is $O(|\mathcal{X}|(|\mathcal{Y}^1| + \cdots + |\mathcal{Y}^k|)$. Considering space complexity, the MfIB algorithm needs to store all the joint distributions $p(X, Y^i)$. Thus, the space complexity is $O(|\mathcal{X}||\mathcal{Y}^1| + \cdots + |\mathcal{X}||\mathcal{Y}^k|)$. This indicates that the time complexity and the space complexity of MfIB algorithm are liner on the input.

4 Experiments

In this section, we evaluate our proposed MfIB algorithm on the unsupervised image categorization task, and show the effectiveness of MfIB.

4.1 Image Data Sets

Seven benchmark image data sets, soccer [Weijer and Schmid, 2006], MSRC [Winn *et al.*, 2005], Sports events [Li and Li, 2007], 17flwoers [Nilsback and Zisserman, 2006], dslr, webcam and amazon [Saenko *et al.*, 2010], are employed to validate MfIB algorithm. The corresponding details are described in Table 1. It should be noted that the categories of the data sets used in this paper vary from 7 to 31, and the sizes of the images vary from 240 to 2813. So the tasks of categorizing them without any supervision are very challenging.

4.2 Image Preprocessing

For data preprocessing, we use the "Bag-of-Words" (BoW) model to represent images in our experiments, which is widely used in the field of unsupervised image classification [Sivic *et al.*, 2005; Lou *et al.*, 2010]. The construction of BoW model can be implemented through three steps, (1) Detecting and representing local patches for each image; (2) Building a visual vocabulary by vector quantization; (3) Mapping the descriptors into the vocabulary and representing each image as a histogram.

Finally, each image is transformed to a feature vector, which contains the occurrence number of the individual visual words in the image. At the first step of BoW model, we need extract local features from the images. There are many local feature extraction techniques to extract multiple cues information (such as color, shape and texture information) from images in the field of computer vision. Choosing one appropriate feature extraction method for the data set is not an easy task. While the color information is crucial to discriminate players from two sport teams, the shape is essential to separate oranges from bananas. In general, people want to combine multiple cues to discriminate the categories [Nilsback and Zisserman, 2006; Fernando *et al.*, 2012]. We adopt the MfIB to combine multiple cues of features. In this framework, one feature variable Y^i is used to represent one cue of features and multiple aspects' information is combined to discriminate the categories. Thus we can use the MfIB algorithm to categorize the images from multiple cues. In this paper, we adopt three techniques SURF [Bay *et al.*, 2008], Color Attention [Khan *et al.*, 2009] and TPLBP [Wolf *et al.*, 2008] to extract local features from images in the cues of local shape, color and texture respectively. Each feature type has its own visual vocabulary with the size of 1000 in the second step of BoW.

4.3 Evaluation Criterion

In this paper, we employ the clustering accuracy (AC) [Cai *et al.*, 2009] to evaluate the performance of different methods, which is defined as:

$$AC = \frac{\sum_{i=1}^{n} \delta(l_i, \mathrm{map}(t_i))}{n}, \tag{9}$$

where t_i denotes the cluster assignment of x_i, l_i is the ground truth label of x_i, and n is size of the data. The delta function $\delta(x, y)$ equals 1 if $x = y$ and equals 0 otherwise. The permutation function $\mathrm{map}(t_i)$ maps each cluster assignment t_i to the equivalent label provided by the data corpus.

4.4 Experimental Results and Analysis

To alleviate the influence caused by random partition, we run each algorithm 10 times, each with a different random initialization. We report the average clustering accuracy and standard deviation. The number of categories M is set to be identical with number of real categories on each data set.

Comparison between original IB and MfIB

The original IB method can only process one feature variable. In this paper, we extend the original IB to MfIB, which can simultaneously process multiple feature variables. In this section, we will do the following experiments to compare the performance of MfIB algorithm with original IB algorithm [Slonim *et al.*, 2002].

- We run the IB algorithm on each type of features and get three results. Each result reflects the patterns extracted from one cue.
- We simply concatenate three types of features as one combined type of features with the vocabulary size of 3000 and run the IB algorithm on the combined features.
- We run the proposed MfIB algorithm simultaneously on three types of features.

Note that, the second comparison is a late feature fusion method [Nilsback and Zisserman, 2006] and the combined features are treated as the instances of one discrete variable in the original IB method. It is different from MfIB.

The evaluation results on the data sets are illustrated in Table 2, from which we have the following observations. (1) The performances of the original IB algorithm on the three individual feature variables are different. The shape feature (SURF) attains the best results on webcam and dslr data sets, the color information performs the best results on soccer and 17flowers data sets, while the MSRC and sports data sets gain the best results by texture features. This phenomenon demonstrates that for different tasks of unsupervised image categorization, none of feature types have the consistent power to perfectly discriminate the categories resided in the images. So, for the task of image categorization, it is not a wise choice to use only one type of features. (2) By simply concatenating three types of features, the average performance is improved compared with one type of features. However, for some data sets, such as MSRC, sports and amazon, the performances on the combining features are either equally matched with or inferior to the performances of IB algorithm on TPLBP features. Therefore, simply combining features can't consistently improve the performance compared with only one type of features. (3) By integrating three types of features, the proposed MfIB algorithm can clearly improve the performances on all data sets compared with the original IB algorithm. Even though the performances of MfIB algorithm and IB algorithm on the combined features are comparatively the same on webcam and dslr data sets, the proposed MfIB algorithm can consistently improve the performance compared with the best results on three individual type features.

Comparison with the state-of-the-art unsupervised image categorization methods

The works presented on [Sivic *et al.*, 2005] and [Lou *et al.*, 2010] have demonstrated the effectiveness of PLSA and IB algorithms on the task of unsupervised image categorization. In this section, we compare our MfIB algorithm with the above two state-of-the-art unsupervised image categorization methods. The comparative results are presented in Table 3, additionally with the results of k-means. From the results, we can see that the MfIB can get more promising results than the state-of-the-art image categorization methods PLSA and IB because our method exploits multiple feature variables simultaneously.

Convergence of MfIB algorithm

Figure 2 shows the repetitions of MfIB algorithm on the seven image data sets. Note that, the values of objective function (5) increase monotonically with each repetition. We also observe that 20 iterations are enough for convergence on all our data sets.

Table 2: The comparison clustering accuracy (%) of MfIB algorithm with the original IB algorithm. Con-Fea denotes the concatenated features.

Data Sets	IB			IB	MfIB
	SURF	ColorAttention	TPLBP	Con-Fea	
Soccer	36.5±1.3	**48.5**±**2.9**	23.0±0.9	**51.1**±**3.0** (↑)	**53.7**±**5.9** (↑)
MSRC	59.4±3.4	46.6±3.2	**70.3**±**5.2**	70.0±3.1 (−)	**76.2**±**1.8** (↑)
Sports	27.8±1.0	29.8±1.5	**50.1**±**3.6**	48.1±3.7 (↓)	**58.2**±**2.5** (↑)
17flowers	29.3±1.0	**32.0**±**2.1**	18.1±0.6	**35.3**±**1.8** (↑)	**38.3**±**1.5** (↑)
Webcam	**42.4**±**2.3**	28.0±1.1	35.7±0.8	**47.1**±**2.9** (↑)	47.9±2.9 (−)
Dslr	**43.6**±**1.5**	34.2±0.8	18.4±1.3	**47.8**±**2.0** (↑)	47.9±1.9 (−)
Amazon	24.5±0.8	12.0±0.4	**27.7**±**1.2**	17.3±0.6 (↓)	**31.2**±**1.0** (↑)
Avg.	37.6	33.0	34.8	45.2 (↑)	**50.5** (↑)

Table 3: The comparison clustering accuracy (%) of MfIB algorithm with the-state-the-art unsupervised image categorization methods. The results of pLSA, kmenas and IB presented in this table are the best results carried out on three individual types of features.

Data Sets	pLSA	kmeans	IB	MfIB
Soccer	47.2±3.5	42.8±4.7	48.5±2.9	**53.7**±**5.9**
MSRC	56.0±5.5	50.5±3.3	70.3±5.2	**76.2**±**1.8**
Sports	41.8±1.4	38.8±2.8	50.1±3.6	**58.2**±**2.5**
17flowers	29.0±0.9	24.5±0.8	32.0±2.1	**38.3**±**1.5**
Webcam	34.3±2.4	30.5±1.1	42.4±2.3	**47.9**±**2.9**
Dslr	33.1±1.5	32.4±1.9	43.6±1.5	**47.9**±**1.9**
Amazon	17.7±1.2	13.3±0.6	27.7±1.2	**31.2**±**1.0**
Avg.	37.0	33.3	44.9	**50.5**

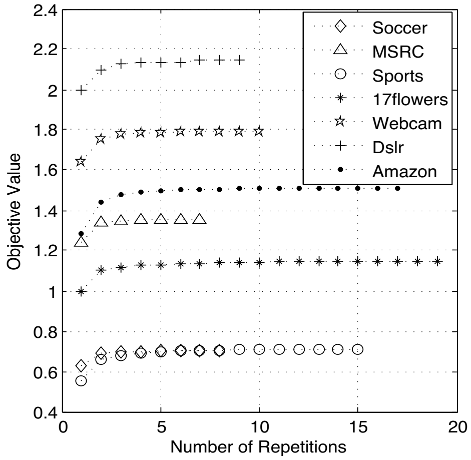


Figure 2: The value of the objective function (5) increases monotonically with the number of repetitions on the run of the data sets.

5 Related Work

This work extends the original IB method [Tishby *et al.*, 1999] to MfIB, which can simultaneously deal with multiple feature variables and analyze the data from multiple feature variables. Multivariate Information Bottleneck [Slonim *et al.*, 2006] is a general principled framework for multivariate extensions of the IB method. However, it merely works on a single modality setting and thus is not adequate for multi-feature clustering task. In our work, we extend the original work on Multivariate IB framework into a multi-feature scenario, which fully takes advantages of the properties of IB method while providing a novel solution to multiple feature clustering task. Besides, the work presented in [Gao *et al.*, 2007] concentrates on multi-view clustering, where each clustering result, generated by one type of features, is treated as one view and the final clustering result is an ensemble results from these views. Differently, our MfIB framework directly takes multiple features as input and tries to fully leverage the correlative information across features.

There are many works in the field of machine learning [Cui *et al.*, 2010] and computer vision [Nilsback and Zisserman, 2006; Fernando *et al.*, 2012] to cope with the problem of multiple feature types. But most of them need the supervision, i.e. the class label information, to help the corresponding algorithms cope with the multiple sources of features. Note that, MfIB is an unsupervised learning method.

6 Conclusions

We have extended the original IB to the MfIB, which aims to extract the data patterns from multiple feature variables. Instead of maximally preserving the information of only one feature variable, the MfIB tries to maintain the information of all the feature variables while X is compressed to T. Therefore, the compressing results can simultaneously reflect the hidden patterns provided by multiple types of features, and the multiple feature variables can complementally help each other to find the patterns which are much closer to the real patterns resided in the data. The experiments on seven challenging benchmark image data sets have confirmed the effectiveness of the proposed MfIB algorithm.

Acknowledgements

This work is supported by the National Natural Science Foundation of China under grant No. 61170223, Joint Funds of the National Natural Science Foundation of China under grant No. U1204610.

References

[Bay *et al.*, 2008] Herbert Bay, Andreas Ess, Tinne Tuytelaars, and Luc Van Gool. Speeded-up robust features (surf). *Computer Vision and Image Understanding*, 110(3):346–359, June 2008.

[Cai *et al.*, 2009] Deng Cai, Xuanhui Wang, and Xiaofei He. Probabilistic dyadic data analysis with local and global consistency. In *Proceedings of the 26th International Conference on Machine Learning*, pages 105–112, Montreal, Canada, June 2009.

[Cover and Thomas, 1991] Thomas M. Cover and Joy A. Thomas. *Elements of Information Theory*. Wiley-Interscience, New York, USA, 1991.

[Cui *et al.*, 2010] Bin Cui, Anthony K. H. Tung, Ce Zhang, and Zhe Zhao. Multiple feature fusion for social media applications. In *Proceedings of the ACM SIGMOD International Conference on Management of Data*, pages 435–446, Indianapolis, USA, June 2010. ACM.

[Dhillon *et al.*, 2004] Inderjit Dhillon, Jacob Kogan, and Charles Nicholas. Feature selection and document clustering. *Survey of Text Mining*, pages 73–100, 2004.

[Fernando *et al.*, 2012] Basura Fernando, Elisa Fromont, Damien Muselet, and Marc Sebban. Discriminative feature fusion for image classification. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 3434–3441, Providence, USA, June 2012. IEEE.

[Gao *et al.*, 2007] Yan Gao, Shiwen Gu, Jianhua Li, and Zhining Liao. The multi-view information bottleneck clustering. In *12th International Conference on Database Systems for Advanced Applications*, pages 912–917, Bangkok, Thailand, April 2007.

[Khan *et al.*, 2009] Fahad Shahbaz Khan, Joost van de Weijer, and Maria Vanrell. Top-down color attention for object recognition. In *IEEE 12th International Conference on Computer Vision*, pages 979–986, Kyoto, Japan, September 2009. IEEE.

[Li and Li, 2007] Li-Jia Li and Fei-Fei Li. What, where and who? classifying events by scene and object recognition. In *IEEE 11th International Conference on Computer Vision*, pages 1–8, Rio de Janeiro, Brazil, October 2007. IEEE.

[Lou *et al.*, 2010] Zhengzheng Lou, Yangdong Ye, and Dong Liu. Unsupervised object category discovery via information bottleneck method. In *Proceedings of the 18th ACM International Conference on Multimedia*, pages 863–866, Firenze, Italy, October 2010. ACM.

[Lowe, 2004] David G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, November 2004.

[Nilsback and Zisserman, 2006] Maria-Elena Nilsback and Andrew Zisserman. A visual vocabulary for flower classification. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1447–1454, New York, USA, June 2006. IEEE.

[Saenko *et al.*, 2010] Kate Saenko, Brian Kulis, Mario Fritz, and Trevor Darrell. Adapting visual category models to new domains. In *Proceedings of the 11th European Conference on Computer Vision*, pages 213–226, Crete, Greece, September 2010.

[Salton, 1991] Gerard Salton. Developments in automatic text retrieval. *Science*, 253(5023):974–980, 1991.

[Sivic *et al.*, 2005] Josef Sivic, Bryan C. Russell, Alexei A. Efros, Andrew Zisserman, and William T. Freeman. Discovering objects and their location in images. In *IEEE 10th International Conference on Computer Vision*, pages 370–377, Beijing, China, October 2005. IEEE.

[Slonim and Tishby, 2000] Noam Slonim and Naftali Tishby. Document clustering using word clusters via the information bottleneck method. In *Proceedings of the 23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 208–215, Athens, Greece, July 2000. ACM.

[Slonim *et al.*, 2002] Noam Slonim, Nir Friedman, and Naftali Tishby. Unsupervised document classification using sequential information maximization. In *Proceedings of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 129–136, Tampere, Finland, August 2002. ACM.

[Slonim *et al.*, 2006] Noam Slonim, Nir Friedman, and Naftali Tishby. Multivariate information bottleneck. *Neural Computation*, 18(8):1739–41789, August 2006.

[Slonim, 2002] Noam Slonim. The informaton bottleneck: Theory and applications. *Doctoral dissertation, The Hebrew University of Jerusalem*, 2002.

[Tishby *et al.*, 1999] Naftali Tishby, Fernando C. Pereira, and William Bialek. The information bottleneck method. In *Proceedings of the 37th Allerton Conference on Communication and Computation*, pages 368–377, Illinois, USA, 1999.

[Tuytelaars *et al.*, 2010] Tinne Tuytelaars, Christoph H. Lampert, Matthew B. Blaschko, and Wray Buntine. Unsupervised object discovery: A comparison. *International Journal of Computer Vision*, 88(2):284–302, June 2010.

[Weijer and Schmid, 2006] Joost Van De Weijer and Cordelia Schmid. Coloring local feature extraction. In *Proceedings of the 9th European Conference on Computer Vision Computer*, pages 334–348, Graz, Austria, May 2006.

[Winn *et al.*, 2005] John Winn, Antonio Criminisi, and Thomas Minka. Object categorization by learned universal visual dictionary. In *IEEE 10th International Conference on Computer Vision*, pages 1800–1807, Beijing, China, October 2005. IEEE.

[Wolf *et al.*, 2008] Lior Wolf, Tal Hassner, and Yaniv Taigman. Descriptor based methods in the wild. In *Faces in Real-Life Images workshop at the European Conference on Computer Vision*, October 2008.