Modeling continuous visual features for semantic image annotation and retrieval

Zhixin Li, Zhiping Shi, Xi Liu, Zhongzhi Shi

Abstract

Automatic image annotation has become an important and challenging problem due to the existence of semantic gap. In this paper, we firstly extend probabilistic latent semantic analysis (PLSA) to model continuous quantity. In addition, corresponding Expectation–Maximization (EM) algorithm is derived to determine the model parameters. Furthermore, in order to deal with the data of different modalities in terms of their characteristics, we present a semantic annotation model which employs continuous PLSA and standard PLSA to model visual features and textual words respectively. The model learns the correlation between these two modalities by an asymmetric learning approach and then it can predict semantic annotation precisely for unseen images. Finally, we compare our approach with several state-of-the-art approaches on the Corel5k and Corel30k datasets. The experiment results show that our approach performs more effectively and accurately.

1. Introduction

Content-based image retrieval (CBIR) has been studied and explored for decades. Its performance, however, is not ideal enough due to the notorious semantic gap (Smeulders et al., 2000). CBIR retrieves images in terms of their visual features, while users often prefer intuitive text-based image searching. Since manual image annotation is expensive and difficult to be extended to large image databases, automatic image annotation has emerged as a striking and crucial problem (Datta et al., 2008).

The state-of-the-art techniques of image auto-annotation can be roughly categorized into two different schools of thought. The first one defines auto-annotation as a traditional supervised classification problem (Chang et al., 2003; Li and Wang, 2003; Cusano et al., 2004; Carneiro et al., 2007), which treats each word (or semantic concept) as an independent class and creates different classifiers for every word. This approach computes similarity at the visual level and annotates a new image by propagating the corresponding words. The second perspective takes a different stand and treats images and texts as equivalent data. It attempts to discover the correlation between visual features and textual words on an unsupervised basis, by estimating the joint distribution of features and words. Thus, it poses annotation as statistical inference in a graphical model. Under this perspective, images are treated as bags of words and features, each of which are assumed generated by a hidden variable. Various approaches differ in the definition of the states of the hidden variable: some associate them with images in the database (Jeon et al., 2003; Lavrenko et al., 2003; Feng et al., 2004), while others associate them with image clusters (Duygulu et al., 2002; Barnard et al., 2003) or latent aspects (topics) (Blei and Jordan, 2003; Monay and Gatica-Perez, 2007; Zhang et al., 2005).

As latent aspect models, PLSA (Hofmann, 2001) and latent Dirichlet allocation (LDA) (Blei et al., 2003) have been successfully applied to annotate and retrieve images. PLSA-WORDS (Monay and Gatica-Perez, 2007) is a representative approach, which achieves the annotation task by constraining the latent space to ensure its consistency in words. However, since standard PLSA can only handle discrete quantity (such as textual words), this approach quantizes feature vectors into discrete visual words for PLSA modeling. Therefore, its annotation performance is sensitive to the clustering granularity. In the area of automatic image annotation, it is generally believed that using continuous feature vectors will give rise to better performance (Lavrenko et al., 2003; Blei and Jordan, 2003; Zhang et al., 2005; Li et al., 2010). In order to model image data precisely, it is required to deal with continuous quantities using PLSA.

This paper proposes continuous PLSA, which assumes that feature vectors in an image are governed by a Gaussian distribution under a given latent aspect other than a multinomial one. In addition, corresponding EM algorithm is derived to estimate the parameters. Then, as general treatment, each image can be treated as a mixture of Gaussians under this model. Furthermore, based on the continuous PLSA and the standard PLSA, we present a semantic
2. Related work

Various approaches based on the classification techniques have been proposed for semantic image annotation and retrieval. A representative work is automatic linguistic indexing of pictures (ALIP) proposed by Li and Wang (2003). ALIP uses two-dimensional multi-resolution hidden Markov models (2D MHMMs) to capture spatial dependencies of visual features of given semantic categories. Besides, the content-based soft annotation (CBSA) system proposed by Chang et al. (2003) is based on binary classifiers trained for each word and it indexes a new image with the output of each classifier. In this stage, these classes all directly compete for the image to annotate. Similar, Blei and Jordan (2003) employ correspondence information for unseen images. We evaluate our approach on Corel datasets and the experiment results show that our approach outperforms several state-of-the-art approaches.

The rest of the paper is organized as follows. Section 2 discusses the related work. Section 3 presents the standard PLSA and continuous PLSA. Furthermore, this section gives the EM algorithm of standard PLSA and derives corresponding EM algorithm of continuous PLSA. Section 4 proposes a semantic annotation model and describes the asymmetric learning algorithm. Experiment results are reported and analyzed in Section 5. Finally, the overall conclusions of this work are presented in Section 6.

3. Standard PLSA and continuous PLSA

Standard PLSA is a statistical latent aspect model for co-occurrence data which associates an unobserved class variable with each observation. However, it can only handle discrete quantity, such as textual words. In this section, we extend standard PLSA to continuous PLSA that can handle continuous quantity, such as visual features. In addition, the corresponding EM algorithm is derived.

3.1. Standard PLSA

Standard PLSA (Hofmann, 2001) introduces a hidden variable \( z_k \) \( (k \in \{1, \ldots, K\}) \) in the generative process of each element \( x_j \) \( (j \in \{1, \ldots, M\}) \) in a document \( d_i \) \( (i \in \{1, \ldots, N\}) \). Given this unobservable variable (latent aspect) \( z_k \), each occurrence \( x_j \) is independent of the document it belongs to, which corresponds to the following joint probability:

\[
P(d_i, z_k, x_j) = P(d_i)P(z_k|d_i)P(x_j|z_k).
\]

The joint probability of the observed variables is obtained by marginalizing over the latent aspect \( z_k \):

\[
P(d_i, x_j) = P(d_i) \sum_{k=1}^{K} P(z_k|d_i)P(x_j|z_k).
\]  

(1)

A representation of the aspect model in terms of a graphical model is depicted in Fig. 1(a). Since the cardinality of the latent aspects is typically smaller than the number of documents (and elements) in the collection, \( K \ll \min(N,M) \), it acts as a bottleneck variable in predicting words.

The model (1) expresses each document as a convex combination of \( K \) aspect vectors. This amounts to matrix decomposition as shown in Fig. 1(b). Essentially, each document is modeled as a mixture of aspects — the histogram for a particular document being composed of a mixture of the histograms corresponding to each aspect.

The model parameters of PLSA are the two conditional distributions: \( P(x_j|z_k) \) and \( P(z_k|d_i) \), which are estimated by an EM algorithm.
on a set of training documents. \( P(x_j|z_k) \) characterizes each aspect and remains valid for documents out of the training set. On the other hand, \( P(z_k|d) \) is only relative to the specific documents and cannot carry any prior information to an unseen document.

An EM algorithm is used to compute the parameters \( P(x_j|z_k) \) and \( P(z_k|d) \) through maximizing the log-likelihood of the observed data
\[
\mathcal{L} = \sum_{i=1}^N \sum_{j=1}^M n(d_i, x_j) \log P(d_i, x_j),
\]
where \( n(d_i, x_j) \) is the count of element \( x_j \) in document \( d_i \). The steps of the EM algorithm are described as follows (Hofmann, 2001):

1. **E-step.** The conditional distribution \( P(z_k|d_i, x_j) \) is computed from the previous estimate of the parameters:
\[
P(z_k|d_i, x_j) = \frac{P(z_k|d_i)P(x_j|z_k)}{\sum_{l=1}^K P(z_l|d_i)P(x_j|z_l)},
\]

2. **M-step.** The parameters \( P(x_j|z_k) \) and \( P(z_k|d_i) \) are updated with the new expected values \( P(z_k|d_i, x_j) \):
\[
P(x_j|z_k) = \frac{\sum_{i=1}^N n(d_i, x_j)P(z_k|d_i, x_j)}{\sum_{m=1}^M \sum_{i=1}^N n(d_i, x_m)P(z_k|d_i, x_m)}, \quad P(z_k|d_i) = \frac{\sum_{j=1}^M n(d_i, x_j)P(z_k|d_i, x_j)}{\sum_{j=1}^M n(d_i, x_j)}.
\]

As for the two parameters, if one parameter \( P(x_j|z_k) \) (or \( P(z_k|d_i) \)) is known, we could quickly infer the other one \( P(z_k|d_i) \) (or \( P(x_j|z_k) \)) using folding-in method — a partial version of the EM algorithm. The method updates the unknown parameters with the known parameters kept fixed, so that it can maximize the likelihood with respect to the previously trained parameters.

### 3.2. Continuous PLSA

Just like standard PLSA, continuous PLSA (Li et al., 2010) is also a statistical latent class model which introduces a hidden variable (latent aspect) \( z_k \) (\( k \in \{1, \ldots, K\} \)) in the generative process of each element \( x_j \) (\( j \in \{1, \ldots, M\} \)) in a document \( d_i \) (\( i \in \{1, \ldots, N\} \)). However, given this unobservable variable \( z_k \), continuous PLSA assumes that elements \( x_j \) are sampled from a multivariate Gaussian distribution, instead of a multinomial one in standard PLSA. Using these definitions, continuous PLSA assumes the following generative process:

1. Select a document \( d_i \) with probability \( P(d_i) \).
2. Sample a latent aspect \( z_k \) with probability \( P(z_k|d_i) \) from a multinomial distribution conditioned on the document \( d_i \).
3. Sample \( x_j \sim P(x_j|z_k) \) from a multivariate Gaussian distribution \( \mathcal{N}(x_j; \mu_k, \Sigma_k) \) conditioned on the latent aspect \( z_k \).

Continuous PLSA has two underlying assumptions. First, the observation pairs \((d_i, x_j)\) are generated independently. Second, the pairs of random variables \((d_i, x_j)\) are conditionally independent given the latent aspect \( z_k \). Thus, the joint probability of the observed variables is obtained by marginalizing over the latent aspect \( z_k \):
\[
P(d_i, x_j) = P(d_i) \sum_{k=1}^K P(z_k|d_i)P(x_j|z_k).
\]

A representation of the model in terms of a graphical model is depicted in Fig. 2.

The mixture of Gaussian is assumed for the conditional probability \( P(x_j|z_k) \). In other words, the elements are generated from \( K \) Gaussian distributions, each one corresponding a \( z_k \). For a specific latent aspect \( z_k \), the condition probability density function of elements \( x_j \) is
\[
P(x_j|z_k) = \frac{1}{(2\pi)^{D/2}\Sigma_k^{1/2}} \exp\left\{ -\frac{1}{2}(x_j - \mu_k)'\Sigma_k^{-1}(x_j - \mu_k) \right\},
\]
where \( D \) is the dimension, \( \mu_k \) is a \( D \)-dimensional mean vector and \( \Sigma_k \) is a \( D \times D \) covariance matrix.

Following the maximum likelihood principle, \( P(z_k|d_i) \) and \( P(x_j|z_k) \) can be determined by maximization of the log-likelihood function
\[
\mathcal{L} = \sum_{i=1}^N \sum_{j=1}^M n(d_i, x_j) \log P(d_i, x_j)
\]
\[
= \sum_{i=1}^N \sum_{j=1}^M n(d_i, x_j) \left[ \log P(d_i) + \log \sum_{k=1}^K P(z_k|d_i)P(x_j|z_k) \right],
\]
where \( n(d_i, x_j) \) denotes the number of element \( x_j \) in document \( d_i \).

The standard procedure for maximum likelihood estimation in latent variable models is the EM algorithm (Dempster et al., 1977; Bishop, 2006). In E-step, applying Bayes’ theorem to (6), one can obtain
\[
P(z_k|d_i, x_j) = \frac{P(z_k|d_i)P(x_j|z_k)}{\sum_{l=1}^K P(z_l|d_i)P(x_j|z_l)}.
\]

In M-step, one has to maximize the expectation of the complete-data log-likelihood
\[
\mathbb{E} [\mathcal{L}] = \sum_{i=1}^N \sum_{j=1}^M n(d_i, x_j) \sum_{k=1}^K P(z_k|d_i, x_j) \log[P(z_k|d_i)|P(x_j|z_k)].
\]
Maximizing (10) with Lagrange multipliers to \( P(z_k|d_i) \) and \( P(x_j|z_k) \) respectively, under the following constraints
\[
\sum_{k=1}^K P(z_k|d_i) = 1, \quad \sum_{k=1}^K P(z_k|d_i, x_j) = 1.
\]

For any \( d_i, z_k \) and \( x_j \), the parameters are determined as
4. Semantic annotation model

We discuss here a model to learn the semantic information from visual and textual modalities of annotated images. In addition, the learning and annotating algorithm is described.

4.1. Gaussian-multinomial PLSA

In order to deal with the data of different modalities in terms of their characteristics, we employ continuous PLSA and standard PLSA to model visual features and textual words respectively. These two models are linked by sharing the same distribution over latent aspects \( P(z|d) \). We refer to this semantic annotation model as Gaussian-multinomial PLSA (GM-PLSA), which is represented in Fig. 3.

GM-PLSA assumes the following generative process:

1. Select a document \( d \), with probability \( P(d) \);
2. Sample a latent aspect \( z \), with probability \( P(z|d) \) from a multinomial distribution conditioned on the document \( d \);
3. For each of the words, sample \( w \), from a multinomial distribution \( \text{Mult}(\theta_k) \) conditioned on the latent aspect \( z \);
4. For each of the feature vectors, sample \( f \), from a multivariate Gaussian distribution \( \mathcal{N}(\mu_k, \Sigma_k) \) conditioned on the latent aspect \( z \).

Under this modeling approach, each image can be viewed as either a mixture of continuous Gaussian in visual modality or a mixture of discrete words in textual modality. Therefore, it can learn the correlation between features and words effectively and predict semantic annotation precisely for an unseen image.

4.2. Learning and annotating

We adopt asymmetric learning approach to estimate the model parameters because an asymmetric learning gives a better control of the respective influence of each modality in the latent space definition (Monay and Gatica-Perez, 2007). In this learning approach, textual modality is firstly chosen to estimate the mixture of aspects in a given document, which constrains the definition of latent space to ensure its consistency in textual words, while retaining the ability to jointly model visual features. The flow of learning and annotating is described in Fig. 4.

In training stage, each training image is processed and represented as a bag of visual features and textual words. The aspect distributions \( P(z|d) \) are firstly learned for all training documents from textual words only. At the same time, the parameter \( P(w|z) \) (i.e. \( \theta_k \)) is determined too. Then, we use folding-in method described in Section 3.2 to infer the parameters \( \mu_k \) and \( \Sigma_k \) for the visual features with the aspect distributions \( P(z|d) \) kept fixed. Consequently, we can get the model parameters \( \theta_k, \mu_k \) and \( \Sigma_k \) which remain valid in images out of the training set. The learning procedure is described in detail in Algorithm 1.

**Algorithm 1.** Estimation of the parameters: \( \theta_k, \mu_k \) and \( \Sigma_k \)

| Input: | Visual features \( f_n \) and textual words \( w_m \) of training images. |
| Output: | Model parameters \( \theta_k, \mu_k \) and \( \Sigma_k \). |
| Process: |
| 1. | random initialize the \( \theta_k, \mu_k \) and \( \Sigma_k \);
| 2. while | increase in the likelihood of validation data \( \Delta L_v > T_v \) do
| (a) [E step] | for \( k \in 1,...,K \) and all \( (d_i, w_j) \) pairs in training documents do compute \( P(z_k|d_i, w_j) \) with Eq. (3); end for |
| (b) [M step] | for \( k \in 1,...,K \) and \( j \in 1,...,M \) do compute \( P(w_j|z_k) \) with Eq. (4); end for |
| (c) compute the likelihood of validation data \( L_v \) with Eq. (2); end while |
| 3. save \( \theta = \{P(w_1|z_1),P(w_2|z_1),...,P(w_M|z_K)\}; |
| 4. initialize \( \mu_k \) and \( \Sigma_k \);
| 5. while | increase in the likelihood of validation data \( \Delta L_v > T_v \) do
| (a) [E step] | for \( k \in 1,...,K \) and all \( (d_i, f_j) \) pairs in training documents do compute \( P(z_k|d_i, f_j) \) with Eq. (9); end for |
| (b) [M step] | for \( k \in 1,...,K, i \in 1,...,N \) and \( j \in 1,...,M \) do compute \( \mu_k \) with Eq. (12); compute \( \Sigma_k \) with Eq. (13); end for |
| (c) compute the likelihood of validation data \( L_v \) with Eq. (8); end while |
| 6. save \( \mu_k \) and \( \Sigma_k \); |

In annotation stage, given visual features of each test image and the previously estimated parameters \( \mu_k \) and \( \Sigma_k \), the aspect distribution \( P(z|d_{new}) \) can be inferred for a new document \( d_{new} \) using the folding-in method. The posterior probability of each word in the vocabulary is then computed by
As usual, we choose five words with the largest posterior probabilities as annotations of an unseen image.

Having estimated the parameters of GM-PLSA, the stage of semantic retrieval can be put into practice directly. The retrieval algorithm takes as inputs a semantic word \( w_q \) and a database of test images. After annotating each image in the test database, the retrieval algorithm ranks the images labeled with the query word by decreasing posterior probability \( P(w_q|d_{new}) \).

Through these algorithms, standard PLSA and continuous PLSA can work cooperatively. Furthermore, they are employed to deal with different modality data properly. Therefore, the task of image annotation and retrieval can be achieved effectively and accurately.

5. Experiment results

We have implemented PLSA-WORDS and our approach in a prototype system. The process of PLSA-based models fitting is executed offline; the task of semantic image annotation and retrieval is performed online.

5.1. Datasets

In order to test the effectiveness and accuracy of the proposed approach, we conduct our experiments on two datasets, i.e. Corel5k and Corel30k.

Corel5k dataset, originally used in (Duygulu et al., 2002), is a basic comparative dataset for recent research works on image annotation. The dataset consists of 5000 images from 50 Corel Stock Photo cds. Each cd includes 100 images on the same topic. We divided this dataset into 3 parts: a training set of 4000 images, a validation set of 500 images and a test set of 500 images. The validation set is used to determine system parameters. After fixing the parameters, we merged the 4000 training set and 500 validation set to form a new training set. This corresponding to the training set of 4500 images and the test set of 500 images used by Duygulu et al. (2002). Furthermore, only the words (260 in total) that are used as annotations for at least 8 images are selected into vocabulary.

Corel30k dataset is an extension of the Corel5k dataset based on a substantially larger database, which tries to correct some of the limitation in Corel5k such as small number of examples and small size of the vocabulary. Corel30k dataset contains 31,695 images and 5587 words. Out of the 31,695 images, 90% are used for model training (28,525) and 10% for testing (3170 images). As in (Carneiro et al., 2007), only the words (950 in total) that are used as annotations for at least 10 images are selected into vocabulary.

5.2. Evaluation measures

Image annotation performance is evaluated by comparing the captions automatically generated for the test set with the human-produced ground truth. Similar to Feng et al. (2004), we define the automatic annotation as the five semantic words of largest posterior probability, and compute the recall and precision of every word in the test set. For a given semantic word, recall = B/C and precision = B/A, where A is the number of images automatically annotated with a given word; B is the number of images correctly annotated with that word; C is the number of images having that word in ground truth annotation. The average word precision and word recall values summarize the system performance.

\[
P(w_m|d_{new}) = \sum_{k=1}^{K} P(z_k|d_{new})P(w_m|z_k).
\]
Besides, the performance of semantic retrieval is also evaluated by measuring precision and recall. These measures, however, are not good enough to evaluate the retrieval performance comprehensively. So we use another important metric called mean average precision (mAP), which has been a standard measure for the retrieval of text document for years. mAP has the ability to summarize the retrieval performance in a meaningful way. To compute it, the AP of a query \( q \) is first defined as the sum of the precisions of the correctly retrieved images at rank \( i \) divided by the total number of relevant images \( r(q) \) for this query,

\[
AP(q) = \frac{\sum_{i=1}^{r(q)} \text{precision}(i)}{r(q)}. \tag{16}
\]

The AP measure of a query is thus sensitive to the entire ranking of documents. The mean of the AP of \( N_q \) queries summarizes the performance of a retrieval system in one mAP value,

\[
mAP = \frac{\sum_{q=1}^{N_q} AP(q)}{N_q}. \tag{17}
\]

### 5.3. Parameters setting

An important parameter of the experiment is the number of latent aspects for the PLSA-based models. Since the number of latent aspects defines the capacity of the model — the number of model parameters, it can determine the training time and system efficiency to a large extent. Through a comprehensive analysis on the experiment results, PLSA-WORDS (Monay and Gatica-Perez, 2007) choose 120 as aspect number. Therefore, we also use 120 latent aspects to construct the annotation model so as to compare GM-PLSA with PLSA-WORDS fairly.

PLSA is known for overfitting data. We control the overfitting of our model by early stopping, in which we do not necessarily achieve the optimization until convergence, but instead stop updating the parameters once the performance on hold-out data is not improving. Our algorithm stops the iterative process based on the likelihood of a validation set. We consider the folding-in likelihood, which allows good performance prediction and overfitting control without the need for a tempered version of the EM algorithm (Brants, 2005). The folding-in likelihood of the validation set is defined as

\[
L_{\text{valid}} = \prod_{i=1}^{N_{\text{valid}}} \prod_{j=1}^{M} \sum_{k=1}^{K} P(z_{ik} | d_{ij}) P(x_{ij} | z_{ik}). \tag{18}
\]

The focus of this paper is not on image feature selection and our approach is independent of visual features. So our prototype system uses similar features to Feng et al. (2004) for easy comparison. We simply decompose images into a set of blocks (the size of each block is empirically determined as \( 32 \times 32 \) through a series of experiments on the validation set), then compute a 36 dimensional
feature vector for each block, consisting of 24 color features (auto correlogram (Huang et al., 1999) computed over 8 quantized colors and 3 Manhattan Distances) and 12 texture features (Gabor energy (Manjunath and Ma, 1996) computed over 3 scales and 4 orientations). As a result, each block is represented as a 36 dimension feature vector. Then each image is represented as a bag of features, that is, a set of 36 dimension vectors. All the feature vectors of training images form the inputs of continuous PLSA. This preprocessing procedure provides an uniform interface for continuous PLSA modeling.

5.4. Results on Corel5k dataset

5.4.1. Results on automatic image annotation

In this section, the performance of our model (GM-PLSA) is compared with several state-of-the-art models — the Co-occurrence model (Mori et al., 1999), the Translation Model (Duygulu et al., 2002), CMRM (Jeon et al., 2003), CRM (Lavrenko et al., 2003), MBRM (Feng et al., 2004), PLSA-WORDS (Monay and Gatica-Perez, 2007) and PLSA-FUSION (Li et al., 2009). We evaluate the performance of image annotation by comparing the captions automatically generated with the original manual annotations. Similarly to Lavrenko et al. (2003), we compute the recall and precision of every word in the test set and use the mean of these values to summarize the system performance.

We report the results on two sets of words: the subset of 49 best words and the complete set of all 260 words that occur in the training set. The systematic evaluation results are shown in Table 1. From the table, we can see that our model performs significantly better than all other models. We believe that using continuous PLSA and standard PLSA to model visual and textual data respectively is the reason for this result.

Several examples of annotation obtained by our prototype system are shown in Fig. 5. Here top five words are taken as annotation of the image. We can see that even the system annotates an image with a word not contained in the ground truth, this annotation is frequently plausible.

5.4.2. Results on semantic image retrieval

In this section, mean average precision (mAP) is employed as a metric to evaluate the performance of single word retrieval. We only compare our model with CMRM, CRM, MBRM, PLSA-WORDS and PLSA-FUSION, because mAP of other models cannot be accessed directly from the literatures.

The annotation results ignore rank order. However, users always like to rank retrieval images and hope that the top ranked ones are relative images. In fact, most users do not want to see more than even 10 or 20 images in a query. Therefore, rank order is very important for image retrieval. Given a query word, our system will return all the images which are automatically annotated with the query word and rank the images according to the posterior probabilities of that word. Table 2 shows that the retrieval performance of GM-PLSA is better than that of other models.

Fig. 6 presents four ranked retrieval results obtained with single word queries for challenging concepts. The diversity of visual appearance of the returned images indicates that our model has good generalization ability.

In summary, the experiment results on Corel5k show that GM-PLSA outperforms several state-of-the-art models in many respects, which proves that the continuous PLSA is effective in modeling visual features.
5.5. Results on Corel30k dataset

The Corel30k dataset provides a much larger database size and vocabulary size compared with Corel5k. Since Corel30k is a new dataset, we only compare our model with PLSA-WORDS.

Fig. 7 presents the precision-recall curves of PLSA-WORDS and GM-PLSA on the Corel30k dataset, with the number of annotations from 2 to 10. The precision and recall values are the mean values calculated over all words. From the figure we can see that GM-PLSA consistently outperforms PLSA-WORDS.

The superior performance of GM-PLSA on precision and recall directly results in its great semantic retrieval performance. From Table 3 we can also see the great improvements of GM-PLSA over PLSA-WORDS.

Overall, the experiments on Corel30k indicate that GM-PLSA is fairly stable with respect to its parameters setting. Moreover, since this annotation model integrate standard PLSA and continuous PLSA, it has better robustness and scalability.

6. Conclusions

In this paper, we have proposed continuous PLSA to model continuous quantity and develop an EM-based iterative procedure to estimate the parameters. Furthermore, we present a semantic annotation model, which employ continuous PLSA and standard PLSA to deal with the visual and textual data respectively. An adaptive asymmetric learning approach is adopted to learn the correlation between visual and textual modalities. Experiments on the Corel dataset prove that our approach is promising for semantic image annotation and retrieval. In comparison to several state-of-the-art annotation models, higher accuracy and superior effectiveness of our approach are reported.

Acknowledgments

This work is supported by the National Basic Research Priorities Programme (No. 2007CB311004), the National Science and Technology Support Plan (No. 2006BAC08B06) and the National Natural Science Foundation of China (Nos. 60933004, 60903141, 60903079, 60775035, 60970088).

References


