Modeling Hidden Topics on Document Manifold*

Deng Cai[†] Qiaozhu Mei[†] Xiaofei He[‡] Jiawei Han[†]

[†] Department of Computer Science, University of Illinois at Urbana-Champaign

[‡] College of Computer Science, Zhejiang University, China

January 2008

Abstract

Topic modeling has been a key problem for document analysis. One of the canonical approaches for topic modeling is Probabilistic Latent Semantic Indexing, which maximizes the joint probability of documents and terms in the corpus. The major disadvantage of PLSI is that it estimates the probability distribution of each document on the hidden topics independently and the number of parameters in the model grows linearly with the size of the corpus, which leads to serious problems with overfitting. Latent Dirichlet Allocation (LDA) is proposed to overcome this problem by treating the probability distribution of each document over topics as a hidden random variable. Both of these two methods discover the hidden topics in the Euclidean space. However, there is no convincing evidence that the document space is Euclidean, or *flat*. Therefore, it is more natural and reasonable to assume that the document space is a manifold, either linear or nonlinear. In this paper, we consider the problem of topic modeling on intrinsic document manifold. Specifically, we propose a novel algorithm called Laplacian Probabilistic Latent Semantic Indexing (LapPLSI) for topic modeling. LapPLSI models the document space as a submanifold embedded in the ambient space and directly performs the topic modeling on this document manifold in question. We compare the proposed LapPLSI approach with PLSI and LDA on three text data sets. Experimental results show that LapPLSI provides better representation in the sense of semantic structure.

*The work was supported in part by the U.S. National Science Foundation NSF IIS-05-13678, NSF BDI-05-15813 and MIAS (a DHS Institute of Discrete Science Center for Multimodal Information Access and Synthesis). Any opinions, findings, and conclusions or recommendations expressed here are those of the authors and do not necessarily reflect the views of the funding agencies.

1 Introduction

Document representation has been a key problem for document analysis and processing, such as clustering, classification and retrieval [7][9][10]. The Vector Space Model (VSM) might be one of the most popular models for document representation. In VSM, each document is represented as a *bag of words*. Correspondingly, the inner product (or, cosine similarity) is used as the standard similarity measure for documents or documents and queries. Unfortunately, it is well known that VSM has severe drawbacks, mainly due to the ambiguity of words (*polysemy*) and the personal style and individual differences in word usage (*synonymy*).

To deal with these problems, IR researchers have proposed several dimensionality reduction techniques, most notably Latent Semantic Indexing (LSI) [7]. LSI uses a Singular Value Decomposition (SVD) of the term-document matrix X to identify a linear subspace (so-called *latent semantic space*) that captures most of the variance in the data set. The general claim is that similarities between documents or between documents and queries can be more reliably estimated in the reduced latent space representation than in the original representation. LSI received a lot of attentions during these years and many variants of LSI have been proposed [1][12][20][21].

Despite its remarkable success in different domains, LSI has a number of deficits, mainly due to its unsatisfactory statistical formulation [11]. To address this issue, Hofmann [10] proposed a generative probabilistic model named Probabilistic Latent Semantic Indexing (PLSI). PLSI models each word in a document as a sample from a mixture model, where the mixture components are multinomial random variables that can be viewed as representations of "topics." Each document is represented as a list of mixing proportions for these mixture components and thereby reduced to a probability distribution on a fixed set of topics. This distribution is the "reduced representation" associated with the document. The major disadvantage of PLSI is that it estimates the probability distribution of each document on the hidden topics independently and the number of parameters in the model grows linearly with the size of the corpus. This leads to serious problems with overfitting [16][5][19]. Latent Dirichlet Allocation (LDA) is then proposed to overcome this problem by treating the probability distribution of each document over topics as a K-parameter hidden random variable rather than a large set of individual parameters, where the K is the number of hidden topics.

Both of the above two topic modeling approaches discover the hidden topics in the Euclidean space. However, there is no convincing evidence that the documents are actually sampled from a Euclidean space. Recent studies suggest that the documents are usually sampled from a nonlinear low-dimensional manifold which is embedded in the high-dimensional ambient space [9][24]. Thus, the local geometric structure is essential to reveal the hidden semantics in the corpora.

In this paper, we propose a new algorithm called **Laplacian Probabilistic Latent Semantic Indexing** (LapPLSI). LapPLSI considers the topic modeling on the document manifold. It models the document space as a submanifold embedded in the ambient space and directly perform the topic modeling on this document manifold in question. By discovering the local neighborhood structure, our algorithm can have more discriminating power than PLSI and LDA. Specifically, LapPLSI first builds an nearest neighbor graph to model the local document manifold structure. It is natural to assume that two sufficiently close documents have similar probability distribution over different topics. The nearest neighbor graph structure is then incorporated into the log-likelihood maximization as a regularization term for LapPLSI. In this way, the topic model estimated by LapPLSI maximizes the joint probability over the corpus and simultaneously respects the local manifold structure.

It is worthwhile to highlight several aspects of our proposed algorithm here:

1. The conventional generative probabilistic modeling approaches, *e.g.*, PLSI and LDA, discover the hidden topics in the Euclidean space. Our approach considers the problem of topic modeling directly on the document manifold in question and discovers the hidden topics.

2. The graph Laplacian used in our algorithm is a discrete approximation to the Laplace-Beltrami operator defined on manifold. By discovering the local neighborhood structure, our algorithm can have more discriminating power than PLSI and LDA.

3. Our algorithm constructs a nearest neighbor graph to model the intrinsic structure in the data, which is unsupervised. When there is network structure available, *e.g*. hyperlink between Web pages, it can be naturally used to construct the graph.

The rest of the paper is organized as follows: in Section 2, we give a brief review of topic modeling with PLSI and LDA. Section 3 introduces our algorithm and give a theoretical analysis of the algorithm. Extensive experimental results on document modeling and document clustering are presented in Section 4. Finally, we provide some concluding remarks and suggestions for future work in Section 5.

2 A brief review of PLSI and LDA

The core of Probabilistic Latent Semantic Indexing (PLSI) is a latent variable model for co-occurrence data which associates an unobserved topic variable $z_k \in \{z_1, \cdots, z_K\}$ with the occurrence of a word $w_j \in \{w_1, \cdots, w_M\}$ in a particular document $d_i \in \{d_1, \cdots, d_N\}$. As a generative model for word/document co-occurrences, PLSI is defined by the following scheme:

1. select a document d_i with probability $P(d_i)$,
2. pick a latent topic z_k with probability $P(z_k|d_i)$,
3. generate a word w_j with probability $P(w_j|z_k)$.

As a result one obtains an observation pair (d_i, w_j), while the latent topic variable z_k is discarded. Translating the data generation process into a joint probability model results in the expression

$$\begin{aligned} P(d_i, w_j) &= P(d_i)P(w_j|d_i), \\ P(w_j|d_i) &= \sum_{k=1}^{K} P(w_j|z_k)P(z_k|d_i). \end{aligned} \tag{1}$$

The parameters can be estimated by maximizing the log-likelihood

$$\begin{aligned} \mathcal{L} &= \sum_{i=1}^{N}\sum_{j=1}^{M} n(d_i, w_j) \log P(d_i, w_j) \\ &\propto \sum_{i=1}^{N}\sum_{j=1}^{M} n(d_i, w_j) \log \sum_{k=1}^{K} P(w_j|z_k)P(z_k|d_i) \end{aligned} \tag{2}$$

where $n(d_i, w_j)$ the number of occurrences of term w_j in document d_i. The above optimization problem can be solved by using standard EM algorithm [8].

Notice that there are $NK + MK$ parameters $\{P(w_j|z_k), P(z_k|d_i)\}$ which are independently estimated in PLSI model. It is easy to see that the number of parameters in PLSI grows linearly with the number of training documents (N). The linear growth in parameters suggests that the model is prone to overfitting [16][5].

To address this issue, Latent Dirichlet Allocation (LDA) [5] is then proposed. LDA assumes that the probability distributions of documents over topics are generated from the same Dirichlet distribution with

K parameters. Essentially, LDA modifies the second step of PLSI generating scheme:

1. select a document d_i with probability $P(d_i)$,
2. pick a latent topic z_k,
    - 2.1 generate $\theta_i \sim Dir(\alpha)$,
    - 2.2 pick a latent topic z_k with probability $P(z_k|\theta_i)$,
3. generate a word w_j with probability $P(w_j|z_k)$.

$Dir(\alpha)$ is the Dirichlet distribution with a K-dimensional parameter α.

The $K + MK$ parameters in a K-topic LDA model do not grow with the size of the corpus. Thus, LDA does not suffer from the same overfitting issue as PLSI.

3 Laplacian Probabilistic Latent Semantic Indexing

By assuming that the probability distributions of documents over topics are generated from the same Dirichlet distribution, LDA avoids the overfitting problem of PLSI. However, both of these two algorithms fail to discover the intrinsic geometrical and discriminating structure of the document spare, which is essential to the real applications. In this Section, we introduce our LapPLSI algorithm which avoids this limitation by incorporating a geometrically based regularizer.

3.1 The Latent Variable Model with Manifold Regularization

Recall that the documents $d \in D$ are drawn according to the distribution P_D. One might hope that knowledge of the distribution P_D can be exploited for better estimation of the conditional distribution $P(z|d)$. Nevertheless, if there is no identifiable relation between P_D and the conditional distribution $P(z|d)$, the knowledge of P_D is unlikely to be very useful.

Therefore, we will make a specific assumption about the connection between P_D and the conditional distribution $P(z|d)$. We assume that if two documents $d_1, d_2 \in D$ are *close* in the *intrinsic* geometry of P_D, then the conditional distributions $P(z|d_1)$ and $P(z|d_2)$ are similar to each other. In other words, the conditional probability distribution $P(z|d)$ varies smoothly along the geodesics in the intrinsic geometry of

P_D. This assumption is also referred to as *manifold assumption* [3], which plays an essential rule in developing various kinds of algorithms including dimensionality reduction algorithms [3][9] and semi-supervised learning algorithms [4][25].

Let $f_k(d) = P(z_k|d)$ be the conditional Probability Distribution Function (PDF), we use $\|f_k\|^2_M$ to measure the smoothness of f_k along the geodesics in the intrinsic geometry of P_D. When we consider the case that the support[1] of P_D is a compact submanifold $\mathcal{M} \subset \mathbb{R}^M$, a natural choice for $\|f_k\|^2_M$ is

$$\|f_k\|^2_M = \int_{d \in \mathcal{M}} \|\nabla_{\mathcal{M}} f_k\|^2 dP_D(d) \tag{3}$$

where $\nabla_{\mathcal{M}}$ is the gradient of f_k along the manifold $\mathcal{M}$ and the integral is taken over the distribution P_D.

In reality, the document manifold is usually unknown. Thus, $\|f_k\|^2_M$ in Eqn. (3) can not be computed. Recent studies on spectral graph theory [6] and manifold learning theory [2] have demonstrated that $\|f_k\|^2_M$ can be discretely approximated through a nearest neighbor graph on a scatter of data points.

Consider a graph with N vertices where each vertex corresponds to a document in the corpus. Define the edge weight matrix W as follows:

$$W_{ij} = \begin{cases} \cos(d_i, d_j), & \text{if } d_i \in N_p(d_j) \text{ or } d_j \in N_p(d_i) \\ 0, & \text{otherwise.} \end{cases} \tag{4}$$

where $N_p(d_i)$ denotes the set of p nearest neighbors of d_i. Define $L = D - W$, where D is a diagonal matrix whose entries are column (or row, since W is symmetric) sums of W, $D_{ii} = \sum_j W_{ij}$. L is called graph Laplacian [6], which is a discrete approximation to the Laplace-Beltrami operator $\triangle_{\mathcal{M}}$ on the manifold [2].

[1] In mathematics, a support of a function f from a set X to the real numbers $\mathbb{R}$ is a subset Y of X such that $f(x)$ is zero for all $x \in X$ that are not in Y.

Thus, the discrete approximation of $\|f_k\|_M^2$ can be computed as follows:

$$\begin{aligned}\mathcal{R}_k &= \frac{1}{2}\sum_{i,j=1}^{N}\left(P(z_k|d_i) - P(z_k|d_j)\right)^2 W_{ij} \\ &= \sum_{i=1}^{N} P(z_k|d_i)^2 D_{ii} - \sum_{i,j=1}^{N} P(z_k|d_i)P(z_k|d_j)W_{ij} \\ &= \mathbf{f}_k^T D\mathbf{f}_k - \mathbf{f}_k^T W\mathbf{f}_k \\ &= \mathbf{f}_k^T L\mathbf{f}_k\end{aligned} \tag{5}$$

where $\mathbf{f}_k = [f_k(d_1), \cdots, f_k(d_M)]^T = [P(z_k|d_1), \cdots, P(z_k|d_M)]^T$. $\mathcal{R}_k$ can be used to measure the smoothness of conditional probability distribution function $P(z_k|d)$ along the geodesics in the intrinsic geometry of the document set. By minimizing $\mathcal{R}_k$, we get a conditional PDF function f_k which is sufficiently smooth on the document manifold. A intuitive explanation of minimizing $\mathcal{R}_k$ is that if two documents d_i and d_j are close (*i.e.* W_{ij} is big), $f_k(d_i)$ and $f_k(d_j)$ are similar to each other.

Now we can define our new latent variable model. The new model adopts the generative scheme of PLSI. It aims to maximize the *regularized* log-likelihood as follows:

$$\begin{aligned}\mathfrak{L} &= \lambda\mathcal{L} - (1-\lambda)\mathcal{R} = \lambda\mathcal{L} - (1-\lambda)\sum_{k=1}^{K}\mathcal{R}_k \\ &\propto \lambda\sum_{i=1}^{N}\sum_{j=1}^{M} n(d_i, w_j)\log\sum_{k=1}^{K} P(w_j|z_k)P(z_k|d_i) \\ &\quad - \frac{1-\lambda}{2}\sum_{k=1}^{K}\sum_{i,j=1}^{N}\left(P(z_k|d_i) - P(z_k|d_j)\right)^2 W_{ij}\end{aligned} \tag{6}$$

where λ is the regularization parameter.

3.2 Model Fitting with Generalized EM

To see how we can estimate the parameters in our LapPLSI model, we first consider the case that $\lambda = 1$. In this case, LapPLSI boils down to the traditional PLSI model.

The standard procedure for maximum likelihood estimation in latent variable models is the Expectation Maximization (EM) algorithm [8]. EM alternates two steps: (i) an expectation (E) step where posterior probabilities are computed for the latent variables, based on the current estimates of the parameters, (ii) a maximization (M) step, where parameters are updated based on maximizing the so-called expected complete data log-likelihood which depends on the posterior probabilities computed in the E-step.

Recall in PLSI, we have $NK + MK$ parameters $\{P(w_j|z_k), P(z_k|d_i)\}$ and the latent variables are the hidden topics z_k. For simplicity, we use Ψ to denote all the $NK + MK$ parameters.

E-step:

The posterior probabilities for the latent variables are $P(z_k|d_i, w_j)$, which can be computed by simply applying Bayes' formula on Eqn. (1):

$$P(z_k|d_i, w_j) = \frac{P(w_j|z_k)P(z_k|d_i)}{\sum_{l=1}^{K} P(w_j|z_l)P(z_l|d_i)} \tag{7}$$

M-step:

With simple derivations [11], one can obtain the relevant part of the expected complete data log-likelihood for PLSI:

$$Q(\Psi) = \sum_{i=1}^{N}\sum_{j=1}^{M} n(d_i, w_j) \sum_{k=1}^{K} P(z_k|d_i, w_j) \log \left[P(w_j|z_k)P(z_k|d_i)\right]$$

Maximizing $Q(\Psi)$ with respect to the parameters Ψ and with the constraints that $\sum_{k=1}^{K} P(z_k|d_i) = 1$ and $\sum_{j=1}^{M} P(w_j|z_k) = 1$, one can obtain the M-step re-estimation equations [11]:

$$P(w_j|z_k) = \frac{\sum_{i=1}^{N} n(d_i, w_j)P(z_k|d_i, w_j)}{\sum_{m=1}^{M}\sum_{i=1}^{N} n(d_i, w_m)P(z_k|d_i, w_m)}, \tag{8}$$

$$P(z_k|d_i) = \frac{\sum_{j=1}^{M} n(d_i, w_j)P(z_k|d_i, w_j)}{n(d_i)}. \tag{9}$$

With a initial random guess of $\{P(w_j|z_k), P(z_k|d_i)\}$, PLSI alternately applies the E-step equation (7) and M-step equations (8, 9) until a termination condition is met.

Our LapPLSI model adopts the same generative scheme as that of PLSI. Thus, LapPLSI has exactly the same E-step as that of PLSI. For the M-step, it can be derived that the relevant part of the expected complete data log-likelihood for LapPLSI is

$$\begin{aligned}\mathcal{Q}(\Psi) &= \lambda Q(\Psi) - (1-\lambda)\mathcal{R} \\ &= \lambda Q(\Psi) - \frac{1-\lambda}{2}\sum_{k=1}^{K}\sum_{i,j=1}^{N}\left(P(z_k|d_i) - P(z_k|d_j)\right)^2 W_{ij}\end{aligned}$$

Since the regularization part R only involves the parameters $P(z_k|d_i)$, we can get the same M-step re-estimation equation for $P(w_j|z_k)$ as in Eqn. (8). However, we do not have a close form re-estimation equation for $P(z_k|d_i)$. In this case, the traditional EM algorithm can not be applied.

In the following, we discuss how to use the generalized EM algorithm (GEM) [14] to maximize the regularized log-likelihood of LapPLSI in Eqn. (6). The major difference between GEM and traditional EM is in the M-step. Instead of finding the *globally optimal* solutions for Ψ which maximize the expected complete data log-likelihood $\mathcal{Q}(\Psi)$ in the M-step of EM algorithm, GEM only needs to find a "better" Ψ. Let Ψ_n denote the parameter values of the previous iteration and Ψ_{n+1} denote the parameter values of the current iteration. The convergence of GEM algorithm only requires that $\mathcal{Q}(\Psi_{n+1}) \geq \mathcal{Q}(\Psi_n)$ [14].

In each M-step, we have parameter values Ψ_n and try to find Ψ_{n+1} which satisfy $\mathcal{Q}(\Psi_{n+1}) \geq \mathcal{Q}(\Psi_n)$. Apparently, $\mathcal{Q}(\Psi_{n+1}) \geq \mathcal{Q}(\Psi_n)$ holds if $\Psi_{n+1} = \Psi_n$.

We have $\mathcal{Q}(\Psi) = \lambda Q(\Psi) - (1-\lambda)\mathcal{R}$. Let us first find $\Psi_{n+1}^{(1)}$ which maximizes $Q(\Psi)$ instead of the whole $\mathcal{Q}(\Psi)$. This can be done by simply applying Eqn. (8) and (9). Clearly, $\mathcal{Q}(\Psi_{n+1}^{(1)}) \geq \mathcal{Q}(\Psi_n)$ does not necessarily hold. We then try to start from $\Psi_{n+1}^{(1)}$ and decrease $\mathcal{R}$, which can be done through Newton-Raphson method [17]. Notice that $\mathcal{R}$ only involves parameters $P(z_k|d_i)$, we only need to update $P(z_k|d_i)_{n+1}$ part in Ψ_{n+1}.

Given a function $f(x)$ and the initial value x_t, the Newton-Raphson updating formula to decrease (or increase) $f(x)$ is as follows:

$$x_{t+1} = x_t - \gamma\frac{f'(x)}{f''(x)} \tag{10}$$

where $0 \leq \gamma \leq 1$ is the step parameter. Since we have

$$\mathcal{R}_k = \frac{1}{2}\sum_{i,j=1}^{N}\left(P(z_k|d_i) - P(z_k|d_j)\right)^2 W_{ij} = \mathbf{f}_k^T L \mathbf{f}_k \geq 0,$$

the Newton-Raphson method will decrease $\mathcal{R}_k$ in each updating step. With $\Psi_{n+1}^{(1)}$ and put $\mathcal{R}_k$ into the

Newton-Raphson updating formula in Eqn. (10), we can get the close form solution for $\Psi_{n+1}^{(2)}$, and then $\Psi_{n+1}^{(3)}, \cdots, \Psi_{n+1}^{(m)}$, where

$$P(z_k|d_i)_{n+1}^{(t+1)} = (1-\gamma)P(z_k|d_i)_{n+1}^{(t)} + \gamma \frac{\sum_{j=1}^{N} W_{ij} P(z_k|d_j)_{n+1}^{(t)}}{\sum_{j=1}^{N} W_{ij}}. \qquad (11)$$

Clearly, $\sum_{k=1}^{K} P(z_k|d_i)_{n+1}^{(t+1)} = 1$ and $P(z_k|d_i)_{n+1}^{(t+1)} \geq 0$ hold in Eqn. (11) as long as $\sum_{k=1}^{K} P(z_k|d_i)_{n+1}^{(t)} = 1$ and $P(z_k|d_i)_{n+1}^{(t)} \geq 0$. Notice that the $P(w_j|z_k)_{n+1}$ part in Ψ_{n+1} will keep the same.

Every iteration of Eqn. (11) makes the topic distribution smoother on the nearest neighbor graph, essentially, smoother on the document manifold. The step parameter γ can be interpreted as a controlling factor of smoothing the topic distribution among the neighbors. When it is set to 1, the new topic distribution of a document is the average of the old distributions from its neighbors. This parameter will affect the convergence speed but not the convergence result.

We continue the iteration of Eqn. (11) until $\mathcal{Q}(\Psi_{n+1}^{(t+1)}) \leq \mathcal{Q}(\Psi_{n+1}^{(t)})$. Then we test whether $\mathcal{Q}(\Psi_{n+1}^{(t)}) \geq \mathcal{Q}(\Psi_n)$. If not, we reject the proposal of $\Psi_{n+1}^{(t)}$, and return the Ψ_n as the result of the M-step, and continue with the next E-step. We summarize the model fitting approach of our LapPLSI by using generalized EM algorithm in Algorithm (1).

4 Applications and Empirical Results

In this section, we evaluate our LapPLSI algorithm in two application domains: topic representation and document clustering.

In all the mixture models, the expected complete log-likelihood of the data has local maxima at the points where all or some of the mixture components are equal to each other. We run the EM algorithm multiple times with random starting points to improve the local maximum of the EM estimates. To make the comparison fair, we use the same starting points for PLSI and LapPLSI.

Throughout our experiments, we empirically set the number of nearest neighbors p to 5, the value of the Newton step parameter γ to 0.1, the value of the regularization parameter λ to 0.001[2].

[2] We set the parameter λ to make the two terms $Q(\Psi)$ and $\mathcal{R}$ in the LapPLSI regularized log-likelihood comparable. In our experiments, $Q(\Psi)$ is around -10^6 and $\mathcal{R}$ is less than 100.

Algorithm 1 Generalized EM for LapPLSI

```
Input: $N$ documents with a vocabulary size $M$
    The number of topics $K$, The number of nearest neighbors $p$
    Regularization parameter $\lambda$, Newton step parameter $\gamma$
    Termination condition value $\theta$
Output: $P(z_k|d_i), P(w_j|z_k),\ \ i = 1, \cdots, N;\ \ \ j = 1, \cdots, M\ \ \ k = 1, \cdots, K$

Compute the the graph matrix $W$ as in Eqn. (4);
Initialize the probability distributions (parameters) $\Psi_0$;
    $\Psi_0 = \{P(z_k|d_i)_0, P(w_j|z_k)_0\}$
$n \leftarrow 0$;
while (true)
    E-step: Compute the posterior probability as in Eqn. (7) ;
    M-step:
    Compute $P(w_j|z_k)_{n+1}$ as in Eqn. (8);
    Compute $P(z_k|d_i)_{n+1}$ as in Eqn. (9);
    $P(z_k|d_i)^{(1)}_{n+1} \leftarrow P(z_k|d_i)_{n+1}$;
    Compute $P(z_k|d_i)^{(2)}_{n+1}$ as in Eqn. (11);
    while $\left(\mathcal{Q}(\Psi^{(2)}_{n+1}) \geq \mathcal{Q}(\Psi^{(1)}_{n+1})\right)$
        $P(z_k|d_i)^{(1)}_{n+1} \leftarrow P(z_k|d_i)^{(2)}_{n+1}$.
        Compute $P(z_k|d_i)^{(2)}_{n+1}$ as in Eqn. (11)
    if $\left(\mathcal{Q}(\Psi^{(1)}_{n+1}) \geq \mathcal{Q}(\Psi_n)\right)$
        $P(z_k|d_i)_{n+1} \leftarrow P(z_k|d_i)^{(1)}_{n+1}$;
    else
        $\Psi_{n+1} \leftarrow \Psi_n$;
    if $(\mathcal{Q}(\Psi_{n+1}) - \mathcal{Q}(\Psi_n) \leq \theta)$
        break;
    $n \leftarrow n + 1$;
return $\Psi_{n+1}$
```

4.1 Document Modeling

In order to visualize the hidden topics discovered by LapPLSI approach, we conduct the following experiment on TREC AP corpus. We use a subset of the TREC AP corpus containing 2,246 newswire articles with 10,473 unique terms[3].

To compare different approaches, we randomly pick four terms (*i.e.*, "film", "school","space" and "computer"), and find four topics that have these four terms as the most representative terms, respectively. That is, for term w_j, we find the topic z_k such that $P(w_j|z_k) \geq P(w_i|z_k), \forall w_i \neq w_j$. In this way, we can compare

[3] This TREC AP subset can be downloaded at
http://www.cs.princeton.edu/∼blei/lda-c/

Table 1: The 15 most representative terms generated by our LapPLSI algorithm for four topics. The terms are selected according to the probability $P(w|z)$.

Topic 1	Topic 2	Topic 3	Topic 4
film	**school**	**space**	**computer**
movie	students	launch	system
films	university	mission	technology
disney	college	shuttle	systems
universal	student	earth	calif
mca	education	nasa	program
brooks	schools	test	programs
theaters	district	scientists	computers
mary	board	pictures	equipment
dog	public	venus	problem
movies	class	spacecraft	personal
yosemite	teachers	engineers	stations
recycling	black	rocket	numbers
screen	professor	project	design
entertainment	teacher	launched	data

Table 2: The 15 most representative terms generated by the PLSI algorithm for four topics. The terms are selected according to the probability $P(w|z)$.

Topic 1	Topic 2	Topic 3	Topic 4
film	**school**	**space**	**computer**
movie	students	venus	time
company	student	earth	two
disney	university	mission	west
last	schools	nasa	show
environmental	education	shuttle	military
mca	board	spacecraft	president
films	teachers	magellan	virginia
universal	college	telescope	virus
years	teacher	two	told
people	high	astronauts	system
town	public	launch	program
year	state	miles	computers
movies	class	hubble	years
say	parents	make	last

different approaches on the same topic and evaluate the terms generated by them that are used to represent this particular topic. Table 1, 2 and 3 show the terms generated by the LapPLSI, PLSI, and LDA algorithms, respectively, to represent the four topics. For all these three algorithms, we need to pre-define the number of hidden topics in the data set. We empirically set it to 100 as suggested in [5].

All the three topic modeling approaches have quite good performance on these four topics. For the first

Table 3: The 15 most representative terms generated by the LDA algorithm for four topics. The terms are selected according to the probability $P(w|z)$.

Topic 1	Topic 2	Topic 3	Topic 4
film	**school**	**space**	**computer**
movie	students	shuttle	says
theater	education	nasa	system
actor	schools	launch	program
musical	university	mission	long
films	college	earth	theyre
actress	student	venus	numbers
best	teachers	spacecraft	years
last	board	two	time
vietnam	teacher	mars	year
new	high	magellan	work
theaters	class	rocket	number
available	parents	telescope	people
star	teaching	flight	digital
academy	officials	astronauts	software

Table 4: Statistics of TDT2 and Reuters corpora.

	TDT2	Reuters
No. docs. used	9394	8067
No. clusters used	30	30
Max. cluster size	1844	3713
Min. cluster size	52	18
Med. cluster size	131	45
Avg. cluster size	313	269

three topics, although different algorithms select slightly different terms, all these terms can describe the corresponding topic to some extent. For the forth topic ("computer"), LapPLSI is slightly better than PLSI and LDA. As can be seen, LapPLSI selects more terms related to "computers" (*e.g.*, technology, equipment) than PLSI and LDA. In the next subsection, we give a quantitative evaluation of these three algorithms on document clustering.

4.2 Document Clustering

Clustering is one of the most crucial techniques to organize the documents in an unsupervised manner. The hidden topics extracted by the topic modeling approaches can be regarded as clusters. The estimated conditional probability density function $P(z_k|d_i)$ can be used to infer the cluster label of each document. In this experiment, we investigate the use of topic modeling approach for text clustering.

Table 5: Clustering performance on TDT2

K	Accuracy (%)						
	PLSI	LDA	LapPLSI	K-means	AA	NC	NMF-NCW
2	91.5	93.1	**99.9**	97.6	93.9	99.8	99.7
3	82.3	88.2	**99.7**	90.4	90.6	97.9	95.9
4	76.1	80.8	**98.3**	86.3	86.4	95.9	93.2
5	70.0	77.6	**97.3**	81.1	81.6	94.7	89.9
6	69.0	73.6	**97.4**	79.2	79.4	93.4	91.4
7	63.7	67.4	**96.5**	73.8	80.4	89.1	85.8
8	59.8	65.1	**96.0**	72.5	73.8	85.0	82.3
9	63.2	66.5	**95.7**	73.6	73.6	86.0	83.9
10	60.7	65.8	**93.8**	72.3	73.5	81.4	82.6
Avg	70.7	75.3	**97.2**	80.8	81.5	91.5	89.4
K	Normalized Mutual Information (%)						
	PLSI	LDA	LapPLSI	K-means	AA	NC	NMF-NCW
2	71.9	79.3	**98.2**	90.3	84.8	97.8	97.5
3	68.2	79.0	**97.6**	84.4	82.6	94.1	90.9
4	65.1	74.2	**94.6**	82.2	78.6	91.0	89.1
5	62.9	71.7	**93.6**	79.2	75.3	90.4	85.6
6	63.6	71.7	**93.8**	79.6	76.0	90.3	88.8
7	60.0	66.6	**92.0**	75.6	76.0	85.1	83.6
8	57.6	64.7	**91.0**	73.6	70.7	81.4	80.7
9	62.7	68.9	**91.7**	77.6	73.8	83.9	83.9
10	61.8	68.7	**90.2**	76.5	73.4	80.5	82.9
Avg	63.8	71.6	**93.6**	79.9	76.8	88.3	87.0

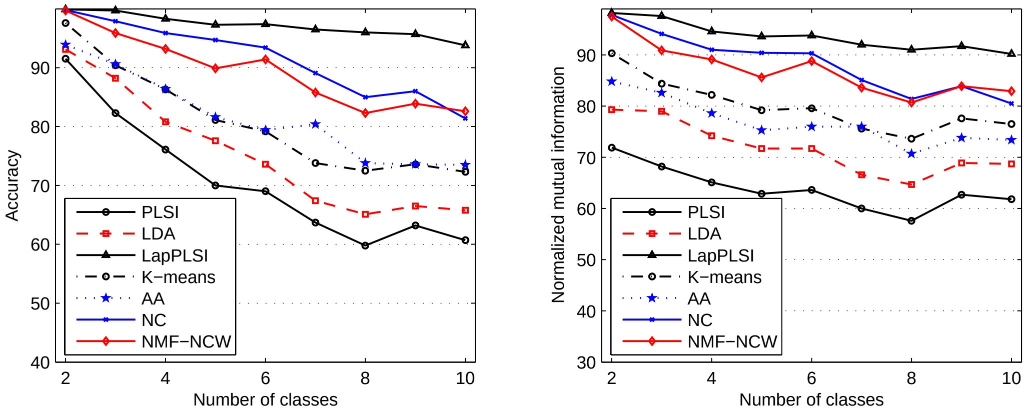

Figure 1: Accuracy and normalized mutual information vs. the number of classes on TDT2 corpus

4.2.1 Data Corpora

We conducted the performance evaluations using the TDT2 [4] and the Reuters[5] document corpora. These two document corpora have been among the ideal test sets for document clustering purposes because documents

[4] Nist Topic Detection and Tracking corpus at http://www.nist.gov/speech/tests/tdt/tdt98/index.htm

in the corpora have been manually clustered based on their topics and each document has been assigned one or more labels indicating which topic/topics it belongs to.

The TDT2 corpus consists of data collected during the first half of 1998 and taken from 6 sources, including 2 newswires (APW, NYT), 2 radio programs (VOA, PRI) and 2 television programs (CNN, ABC). It consists of 11201 on-topic documents which are classified into 96 semantic categories. In this experiment, those documents appearing in two or more categories were removed, and only the largest 30 categories were kept, thus leaving us with 9,394 documents in total.

The Reuters corpus contains 21578 documents which are grouped into 135 clusters. Compared with TDT2 corpus, the Reuters corpus is more difficult for clustering. In TDT2, the content of each cluster is narrowly defined, whereas in Reuters, documents in each cluster have a broader variety of content. Moreover, the Reuters corpus is much more unbalanced, with some large clusters more than 200 times larger than some small ones. In our test, we discarded documents with multiple category labels, and only selected the largest 30 categories. This left us with 8067 documents in total. Table 4 provides the statistics of the two document corpora.

4.2.2 Evaluation Metric

The clustering result is evaluated by comparing the obtained label of each document with that provided by the document corpus. Two metrics, the accuracy (AC) and the normalized mutual information metric ($\overline{MI}$) are used to measure the clustering performance [22]. Given a document $\mathbf{x}_i$, let r_i and s_i be the obtained cluster label and the label provided by the corpus, respectively. The AC is defined as follows:

$$AC = \frac{\sum_{i=1}^{n} \delta(s_i, map(r_i))}{n}$$

where n is the total number of documents and $\delta(x, y)$ is the delta function that equals one if $x = y$ and equals zero otherwise, and map(r_i) is the permutation mapping function that maps each cluster label r_i to the equivalent label from the data corpus. The best mapping can be found by using the Kuhn-Munkres algorithm [13].

Let C denote the set of clusters obtained from the ground truth and C' obtained from our algorithm.

[5] Reuters-21578 corpus is at
http://www.daviddlewis.com/resources/testcollections/reuters21578/

Their mutual information metric $MI(C, C')$ is defined as follows:

$$MI(C, C') = \sum_{c_i \in C, c'_j \in C'} p(c_i, c'_j) \cdot log_2 \frac{p(c_i, c'_j)}{p(c_i) \cdot p(c'_j)}$$

where $p(c_i)$ and $p(c'_j)$ are the probabilities that a document arbitrarily selected from the corpus belongs to the clusters c_i and c'_j, respectively, and $p(c_i, c'_j)$ is the joint probability that the arbitrarily selected document belongs to the clusters c_i as well as c'_j at the same time. In our experiments, we use the normalized mutual information $\overline{MI}$ as follows:

$$\overline{MI}(C, C') = \frac{MI(C, C')}{\max(H(C), H(C'))}$$

where $H(C)$ and $H(C')$ are the entropies of C and C', respectively. It is easy to check that $\overline{MI}(C, C')$ ranges from 0 to 1. $\overline{MI} = 1$ if the two sets of clusters are identical, and $\overline{MI} = 0$ if the two sets are independent.

4.2.3 Performance Evaluations and Comparisons

To demonstrate how the document clustering performance can be improved by topic modeling approaches, we implemented four state-of-the-art clustering algorithms as follows.

- Canonical K-means clustering method (K-means in short).
- Two representative spectral clustering methods: Average Association (AA in short) [23], and Normalized Cut (NC in short) [18][15]. Spectral clustering methods have recently emerged as one of the most effective document clustering tools. These methods are based on graph partitioning theories. They model the given document set using a undirected graph in which each node represents a document, and each edge is assigned a weight to reflect the similarity between two documents. The clustering task is accomplished by finding the best cut of the graph with respect to the predefined criterion function. The difference between AA and NC is the different cut criteria they used. Interestingly, Zha *et al*. [23] has shown that the AA criterion is equivalent to that of the LSI followed by the K-means clustering method if the inner product is used to measure the document similarity.
- Nonnegative Matrix Factorization (NMF) based clustering. We implemented a normalized cut weighted version of NMF (NMF-NCW in short) [22], which has been shown to be a very effective document

Table 6: Clustering performance on Reuters

K	Accuracy (%)						
	PLSI	LDA	LapPLSI	K-means	AA	NC	NMF-NCW
2	72.6	79.1	**92.2**	79.0	82.2	86.3	87.0
3	65.8	69.2	**89.1**	68.7	73.0	78.7	77.6
4	56.7	59.6	**80.3**	62.2	63.1	74.5	74.5
5	52.1	56.5	**73.9**	59.6	59.2	72.0	71.4
6	52.6	53.2	**71.9**	59.7	58.6	70.2	68.7
7	45.5	47.3	**69.3**	53.8	54.0	64.1	63.6
8	45.6	46.8	**65.6**	50.2	47.6	59.9	54.1
9	41.1	42.2	**61.0**	44.5	43.2	57.6	52.8
10	43.0	44.5	**59.4**	47.0	44.2	57.0	53.3
Avg.	52.8	55.4	**73.6**	58.3	58.3	68.9	67.0
K	Normalized Mutual Information (%)						
	PLSI	LDA	LapPLSI	K-means	AA	NC	NMF-NCW
2	23.4	39.1	**70.9**	39.7	45.0	56.3	55.7
3	29.2	39.5	**66.4**	44.5	43.6	54.2	54.3
4	31.4	38.8	**59.4**	47.3	43.5	52.5	56.2
5	35.1	42.3	**59.3**	51.1	48.5	56.7	58.6
6	37.3	43.9	58.1	54.3	50.7	56.4	**59.2**
7	34.6	39.2	**55.8**	49.7	45.1	50.8	53.4
8	34.2	37.2	**49.3**	47.2	42.7	47.6	46.5
9	31.1	34.6	**46.6**	42.5	37.7	44.1	45.1
10	35.1	38.5	**49.5**	47.4	42.6	48.3	49.1
Avg	32.4	39.2	**57.3**	47.1	44.4	51.9	53.1

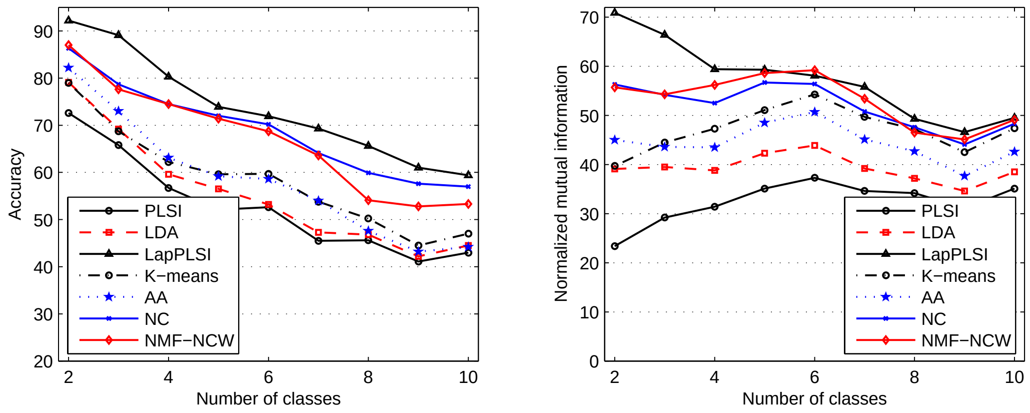

Figure 2: Accuracy and normalized mutual information vs. the number of classes on Reuters corpus

clustering method.

Table 5 and 6 show the evaluation results using the TDT2 and the Reuters corpus, respectively. The evaluations were conducted with the cluster numbers ranging from two to ten. For each given cluster number

K, 50 test runs were conducted on different randomly chosen clusters, and the final performance scores were obtained by averaging the scores from the 50 tests.

These experiments reveal a number of interesting points:

- The LDA approach consistently outperforms PLSI. By assuming that the probability distributions of documents over topics are generated from the same Dirichlet distribution, LDA avoids the overfitting problem of PLSI. This can be observed from our experimental results. However, both of these two topic modeling approaches fail to outperform those standard clustering methods, especially comparing with NC and NMF-NCW. One reason is that both PLSI and LDA discover the hidden topics in the Euclidean space and fail to consider the discriminant structure.

- Our LapPLSI approach gets significantly better performance than PLSI and LDA. Moreover, LapPLSI can even achieve better results than the state-of-the-art clustering algorithms. This shows that by considering the intrinsic geometrical structure of the document space and directly performing topic modeling on this document manifold, LapPLSI can have better hidden topic modeling power in the sense of semantic structure.

- The improvement of LapPLSI over other methods is more significant on the TDT2 corpus than the Reuters corpus. One possible reason is that the document clusters in TDT2 are generally more compact and focused than the clusters in Reuters. Thus, the nearest neighbor graph constructed over TDT2 can better capture the geometrical structure of the document space.

5 Conclusions and Future Work

We have presented a novel method for topic modeling, called Laplacian Probabilistic Latent Semantic Indexing (LapPLSI). LapPLSI models the document space as a submanifold embedded in the ambient space and directly performs the topic modeling on this document manifold in question. As a result, LapPLSI can have more discriminating power than traditional topic modeling approaches which discover the hidden topics in the Euclidean space, *e.g.* PLSI and LDA. Experimental results on document modeling and document clustering show that LapPLSI provides better representation in the sense of semantic structure.

Several questions remain to be investigated in our future work:

1. There is a parameter λ which controls the smoothness of our LapPLSI model. LapPLSI boils down to original PLSI when $\lambda = 1$. Also, it is easy to see that $P(z_k|d_i)$ will be the same for all the documents when $\lambda = 0$. Thus, a suitable value of λ is critical to our algorithm. It remains unclear how to do model selection theoretically and efficiently.

2. We consider the topic modeling on document manifold and develop our approach based on PLSI. The idea of exploiting manifold structure can also be naturally incorporated into other topic modeling algorithms, *e.g.*, Latent Dirichlet Allocation.

3. It would be very interesting to explore different ways of constructing the document graph to model the semantic structure in the data. There is no reason to believe that the nearest neighbor graph is the only or the most natural choice. For example, for web page data it may be more natural to use the hyperlink information to construct the graph.

References

[1] R. Ando. Latent semantic space: Iterative scaling improves precision of inter-document similarity measurement. In *Proc. 2000 Int. Conf. on Research and Development in Information Retrieval (SIGIR'00)*, Athens, Greece, July 2000.

[2] M. Belkin. *Problems of Learning on Manifolds*. PhD thesis, University of Chicago, 2003.

[3] M. Belkin and P. Niyogi. Laplacian eigenmaps and spectral techniques for embedding and clustering. In *Advances in Neural Information Processing Systems 14*, pages 585–591. MIT Press, Cambridge, MA, 2001.

[4] M. Belkin, P. Niyogi, and V. Sindhwani. Manifold regularization: A geometric framework for learning from examples. *Journal of Machine Learning Research*, 2006.

[5] D. Blei, A. Ng, and M. Jordan. Latent dirichlet allocation. *Journal of machine Learning Research*, 2003.

[6] F. R. K. Chung. *Spectral Graph Theory*, volume 92 of *Regional Conference Series in Mathematics*. AMS, 1997.

[7] S. C. Deerwester, S. T. Dumais, T. K. Landauer, G. W. Furnas, and R. A. harshman. Indexing by latent semantic analysis. *Journal of the American Society of Information Science*, 41(6):391–407, 1990.

[8] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)*, 39(1):1–38, 1977.

[9] X. He, D. Cai, H. Liu, and W.-Y. Ma. Locality preserving indexing for document representation. In *Proc. 2004 Int. Conf. on Research and Development in Information Retrieval (SIGIR'04)*, pages 96–103, Sheffield, UK, July 2004.

[10] T. Hofmann. Probabilistic latent semantic indexing. In *Proc. 1999 Int. Conf. on Research and Development in Information Retrieval (SIGIR'99)*, pages 50–57, Berkeley, CA, Aug. 1999.

[11] T. Hofmann. Unsupervised learning by probabilistic latent semantic analysis. *Machine Learning*, 42(1-2):177–196, 2001.

[12] E. Kokiopoulou and Y. Saad. Polynomial filtering in latent semantic indexing for information retrieval. In *Proc. 2004 Int. Conf. on Research and Development in Information Retrieval (SIGIR'04)*, pages 104–111, Sheffield, UK, July 2004.

[13] L. Lovasz and M. Plummer. *Matching Theory*. Akad$\acute{e}$miai Kiad$\acute{o}$, North Holland, Budapest, 1986.

[14] R. Neal and G. Hinton. A view of the EM algorithm that justifies incremental, sparse, and other variants. In *Learning in Graphical Models*. Kluwer, 1998.

[15] A. Y. Ng, M. Jordan, and Y. Weiss. On spectral clustering: Analysis and an algorithm. In *Advances in Neural Information Processing Systems 14*, pages 849–856. MIT Press, Cambridge, MA, 2001.

[16] A. Popescul, L. Ungar, D. Pennock, and S. Lawrence. Probabilistic models for unified collaborative and content-based recommendation in sparse-data environments. In *17th Conference on Uncertainty in Artificial Intelligence*, pages 437–444, 2001.

[17] W. H. Press, B. P. Flannery, S. A. Teukolsky, and W. T. Vetterling. *Numerical Recipes in C: The Art of Scientific Computing*. Cambridge University Press, 1992.

[18] J. Shi and J. Malik. Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):888–905, 2000.

[19] L. Si and R. Jin. Adjusting mixture weights of gaussian mixture model via regularized probabilistic latent semantic analysis. In *The Ninth Pacific-Asia Conference on Knowledge Discovery and Data Mining (PAKDD'05)*, 2005.

[20] C. Tang, S. Dwarkadas, and Z. Xu. On scaling latent semantic indexing for large peer-to-peer systems. In *Proc. 2004 Int. Conf. on Research and Development in Information Retrieval (SIGIR'04)*, pages 112–121, Sheffield, UK, July 2004.

[21] X. Wang, J.-T. Sun, Z. Chen, and C. Zhai. Latent semantic analysis for multiple-type interrelated data objects. In *Proc. 2006 Int. Conf. on Research and Development in Information Retrieval (SIGIR'06)*, pages 236–243, 2006.

[22] W. Xu, X. Liu, and Y. Gong. Document clustering based on non-negative matrix factorization. In *Proc. 2003 Int. Conf. on Research and Development in Information Retrieval (SIGIR'03)*, pages 267–273, Toronto, Canada, Aug. 2003.

[23] H. Zha, C. Ding, M. Gu, X. He, , and H. Simon. Spectral relaxation for k-means clustering. In *Advances in Neural Information Processing Systems 14*, pages 1057–1064. MIT Press, Cambridge, MA, 2001.

[24] D. Zhang, X. Chen, and W. S. Lee. Text classification with kernels on the multinomial manifold. In *SIGIR '05: Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 266–273, 2005.

[25] X. Zhu and J. Lafferty. Harmonic mixtures: combining mixture models and graph-based methods for inductive and scalable semi-supervised learning. In *ICML '05: Proceedings of the 22nd international conference on Machine learning*, pages 1052–1059, 2005.