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.

Categories and Subject Descriptors

G.3 [Mathematics of Computing]: Probability and Statistics—Statistical computing; H.3.1 [Information Storage and Retrieval]: Content Analysis and Indexing—Indexing methods

1. INTRODUCTION

Document representation has been a key problem for document analysis and processing[8][10][11]. 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) [8]. 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][20].

Despite its remarkable success in different domains, LSI has a number of deficits, mainly due to its unsatisfactory statistical formulation [12]. To address this issue, Hofmann [11] 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 [10][23]. 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

\[ P(d_i, w_j) = P(d_i)P(w_j|d_i), \]

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

The parameters can be estimated by maximizing the log-likelihood

\[ \mathcal{L} = \sum_{i=1}^{N} \sum_{j=1}^{M} n(d_i, w_j) \log P(d_i, w_j) \]

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

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 [9].

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 $\alpha$.

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][10] and semi-supervised learning algorithms [4][24].

Let \( f_k(d) = P(z_k|d) \) be the conditional Probability Distribution Function (PDF), we use \( \|f_k\|_2^2 \) 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 \( \|f_k\|_2^2 \) is

\[
\|f_k\|_2^2 = \int_{d \in \mathcal{M}} \|\nabla_{\mathcal{M}} f_k\|^2 dP_D(d) \tag{3}
\]

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, \( \|f_k\|_2^2 \) in Eqn. (3) cannot be computed. Recent studies on spectral graph theory [7] and manifold learning theory [2] have demonstrated that \( \|f_k\|_2^2 \) 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 [7], which is a discrete approximation to the Laplace-Beltrami operator \( \triangle_{\mathcal{M}} \) on the manifold [2]. Thus, the discrete approximation of \( \|f_k\|_2^2 \) can be computed as follows:

\[
\mathcal{R}_k = \frac{1}{2} \sum_{i,j=1}^{N} (P(z_k | d_i) - P(z_k | d_j))^2 W_{ij}
\]

\[
= \sum_{i=1}^{N} P(z_k | d_i)^2 D_{ii} - \sum_{i,j=1}^{N} P(z_k | d_i) P(z_k | d_j) W_{ij} \tag{5}
\]

\[
= \mathbf{t}_k^T D \mathbf{t}_k - \mathbf{t}_k^T L \mathbf{t}_k,
\]

where \( \mathbf{t}_k = [f_k(d_1), \ldots, f_k(d_M)]^T = [P(z_k | d_1), \ldots, P(z_k | d_M)]^T \). \( \mathcal{R}_k \) 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 manifold. By minimizing \( \mathcal{R}_k \), we get a conditional PDF function \( f_k \) which is sufficiently smooth on the document manifold. An 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:

\[
\mathcal{L} = \mathcal{L} - \lambda \mathcal{R} = \mathcal{L} - \frac{1}{2} \sum_{k=1}^{K} \mathcal{R}_k
\]

\[
= \frac{1}{2} \sum_{k=1}^{K} \left( \sum_{i,j=1}^{N} n(d_j, w_i) \right) \log P(w_i | z_k) P(z_k | d_i)
\]

\[
- \frac{1}{2} \sum_{k=1}^{K} \sum_{i,j=1}^{N} (P(z_k | d_i) - P(z_k | d_j))^2 W_{ij}
\]

where \( \lambda \) 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 \( \lambda = 0 \). 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 [9]. 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 \( \Psi \) 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):

\[
P(z_k | d_i, w_j) = \frac{P(w_j | z_k, d_i) P(z_k | d_i)}{P(w_j | z_k)} \tag{7}
\]

**M-step:**

With simple derivations [12], one can obtain the relevant part of the expected complete data log-likelihood for PLSI:

\[
Q(\Psi) = \sum_{i=1}^{N} \sum_{j=1}^{M} n(d_i, w_j) \sum_{k=1}^{K} P(z_k | d_i, w_j) \log P(w_j | z_k) P(z_k | d_i) \tag{8}
\]

Maximizing \( Q(\Psi) \) with respect to the parameters \( \Psi \) 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 [12]:

\[
P(w_j | z_k) = \frac{\sum_{i=1}^{N} n(d_i, w_j) P(z_k | d_i, w_j)}{\sum_{m=1}^{M} \sum_{i=1}^{N} n(d_i, w_m) P(z_k | d_i, w_m)} \tag{9}
\]

With an 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

\[
Q(\Psi) = Q(\Psi) - \lambda \mathcal{R}
\]

\[
= Q(\Psi) - \frac{1}{2} \sum_{k=1}^{K} \sum_{i,j=1}^{N} (P(z_k | d_i) - P(z_k | d_j))^2 W_{ij}
\]
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 cannot 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 $\Psi$ which maximize the expected complete data log-likelihood $Q(\Psi)$ in the M-step of EM algorithm, GEM only needs to find a “better” $\Psi$. Let $\Psi_n$ denote the parameter values of the previous iteration and $\Psi_{n+1}$ denote the parameter values of the current iteration. The convergence of GEM algorithm only requires that $Q(\Psi_{n+1}) \geq Q(\Psi_n)$ [14].

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

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

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:

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

(10)

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

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

the Newton-Raphson method will decrease $\mathcal{R}_k$ in each updating step. With $\Psi^{(1)}_{n+1}$ and put $\Psi_n$ into the Newton-Raphson updating formula in Eqn. (10), we can get the close form solution for $\Psi^{(2)}_{n+1}$, and then $\Psi^{(3)}_{n+1}, \ldots, \Psi^{(m)}_{n+1}$, where

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

(11)

Clearly, $\sum_{k=1}^{K} P(z_k | d_i)_{n+1}^{(1)} = 1$ and $P(z_k | d_i)_{n+1}^{(1)} \geq 0$ hold in Eqn. (11) as long as $\sum_{k=1}^{K} P(z_k | d_i)^{(1)}_{n+1} = 1$ and $P(z_k | d_i)^{(1)}_{n+1} \geq 0$. Notice that the $P(w_j | z_k)_{n+1}$ part in $\Psi_{n+1}$ 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 $\gamma$ 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 $Q(\Psi_{n+1}^{(f+1)}) \leq Q(\Psi_{n+1}^{(f)})$. Then we test whether $Q(\Psi_{n+1}^{(f+1)}) \geq Q(\Psi_{n+1})$. If not, we reject the proposal of $\Psi_{n+1}^{(f+1)}$, and return the $\Psi_n$ 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).

**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:** $P(z_k | d_i), P(w_j | z_k), i = 1, \ldots, N; j = 1, \ldots, M$

1: Compute the the graph matrix $W$ as in Eqn. (4);
2: Initialize the probability distributions (parameters) $\Psi_0$;
3: $n \leftarrow 0$;
4: **while** (true)
5: **E-step:** Compute the posterior probability as in Eqn. (7);
6: **M-step:**
7: Compute $P(z_k | d_i)_{n+1}$ as in Eqn. (8);
8: $P(z_k | d_i)_{n+1}^{(1)} \leftarrow P(z_k | d_i)_{n+1}$;
9: Compute $P(z_k | d_i)_{n+1}^{(2)}$ as in Eqn. (9);
10: while ($Q(\Psi) \geq Q(\Psi^{(n+1)})$
11: $P(z_k | d_i)_{n+1}^{(2)} \leftarrow P(z_k | d_i)_{n+1}$;
12: Compute $P(z_k | d_i)_{n+1}^{(2)}$ as in Eqn. (11);
13: if ($Q(\Psi^{(n+1)}) \geq Q(\Psi_n)$
14: $P(z_k | d_i)_{n+1}^{(1)} \leftarrow P(z_k | d_i)_{n+1}$;
15: else
16: $\Psi_{n+1} \leftarrow \Psi_n$;
17: if ($Q(\Psi_{n+1}) - Q(\Psi_n) \leq \theta$)
18: break;
19: $n \leftarrow n + 1$;
20: **return** $\Psi_{n+1}$

## 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.

We empirically set the number of nearest neighbors $p$ to 7, the value of the Newton step parameter $\gamma$ to 0.1, the value of the regularization parameter $\lambda$ to 1000.

### 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.

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 $P(w_j | z_k) \geq P(w_j | z_{k'}, \forall w_{k'} \neq w_j$. In this way, we can compare different approaches on the same topic and evaluate the terms gen-

---

2This TREC AP subset can be downloaded at [http://www.cs.princeton.edu/~blei/lda-c/](http://www.cs.princeton.edu/~blei/lda-c/)
4.2 Document Clustering

In this section, we give a quantitative evaluation of these three algorithms. Let us denote the corresponding topic to some extent. For the forth algorithm, selecting slightly different terms, all these terms can describe the hidden topics in the data set. We empirically set it to 100 as suggested in [5].

For all three algorithms, we need to pre-define the number of hidden topics. For all 1, 2 and 3 show the terms generated by the LapPLSI, PLSI, and LDA algorithms, respectively, to represent the four topics. For all the three topic modeling approaches have quite good performance on these four topics. For the first 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.

### 4.2.1 Data Corpora

We conducted the performance evaluations using the TDT2 \(^3\) and the Reuters\(^4\) document corpora. These two document corpora have been among the ideal test sets for document clustering because documents 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 has a broader variety of content. Moreover, the Reuters corpus is more unbalanced, with some large clusters more than 200 times larger than some small ones. In our test, we discarded documents appearing in two or more categories, and only the largest 30 categories were kept, thus leaving us with 9,394 documents in total.

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

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

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

<table>
<thead>
<tr>
<th>Topic 1</th>
<th>Topic 2</th>
<th>Topic 3</th>
<th>Topic 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>film</td>
<td>school</td>
<td>space</td>
<td>computer</td>
</tr>
<tr>
<td>movie</td>
<td>students</td>
<td>venus</td>
<td>time</td>
</tr>
<tr>
<td>company</td>
<td>student</td>
<td>earth</td>
<td>two</td>
</tr>
<tr>
<td>disney</td>
<td>university</td>
<td>mission</td>
<td>west</td>
</tr>
<tr>
<td>last</td>
<td>schools</td>
<td>NASA</td>
<td>show</td>
</tr>
<tr>
<td>environmental</td>
<td>education</td>
<td>shuttle</td>
<td>military</td>
</tr>
<tr>
<td>mca</td>
<td>board</td>
<td>magellan</td>
<td>Virginia</td>
</tr>
<tr>
<td>films</td>
<td>teachers</td>
<td>spacecraft</td>
<td>virus</td>
</tr>
<tr>
<td>universal</td>
<td>college</td>
<td>telescope</td>
<td>two</td>
</tr>
<tr>
<td>years</td>
<td>people</td>
<td>teacher</td>
<td>two</td>
</tr>
<tr>
<td>people</td>
<td>town</td>
<td>high</td>
<td>two</td>
</tr>
<tr>
<td>year</td>
<td>movies</td>
<td>public</td>
<td>space</td>
</tr>
<tr>
<td>say</td>
<td>moves</td>
<td>state</td>
<td>launch</td>
</tr>
<tr>
<td></td>
<td></td>
<td>class</td>
<td>months</td>
</tr>
<tr>
<td></td>
<td></td>
<td>parents</td>
<td>make</td>
</tr>
</tbody>
</table>

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

<table>
<thead>
<tr>
<th>Topic 1</th>
<th>Topic 2</th>
<th>Topic 3</th>
<th>Topic 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>film</td>
<td>school</td>
<td>space</td>
<td>computer</td>
</tr>
<tr>
<td>movie</td>
<td>students</td>
<td>Shuttle</td>
<td>Says</td>
</tr>
<tr>
<td>theater</td>
<td>education</td>
<td>NASA</td>
<td>System</td>
</tr>
<tr>
<td>actor</td>
<td>schools</td>
<td>launch</td>
<td>Program</td>
</tr>
<tr>
<td>musical</td>
<td>universities</td>
<td>mission</td>
<td>Long</td>
</tr>
<tr>
<td>films</td>
<td>college</td>
<td>earth</td>
<td>Theyre</td>
</tr>
<tr>
<td>actress</td>
<td>student</td>
<td>Venus</td>
<td>Numbers</td>
</tr>
<tr>
<td>best</td>
<td>teachers</td>
<td>spacecraft</td>
<td>Years</td>
</tr>
<tr>
<td>last</td>
<td>board</td>
<td>two</td>
<td>Time</td>
</tr>
<tr>
<td>vietnam</td>
<td>high</td>
<td>Magellan</td>
<td>Year</td>
</tr>
<tr>
<td>available</td>
<td>theaters</td>
<td>class</td>
<td>rocket</td>
</tr>
<tr>
<td>star</td>
<td>teaching</td>
<td>parents</td>
<td>number</td>
</tr>
<tr>
<td>academy</td>
<td>officials</td>
<td>aerospace</td>
<td>people</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>digital</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>software</td>
</tr>
</tbody>
</table>

### Table 4: Statistics of TDT2 and Reuters corpora.

<table>
<thead>
<tr>
<th></th>
<th>TDT2</th>
<th>Reuters</th>
</tr>
</thead>
<tbody>
<tr>
<td>No. docs used</td>
<td>9394</td>
<td>8067</td>
</tr>
<tr>
<td>No. clusters used</td>
<td>30</td>
<td>30</td>
</tr>
<tr>
<td>Max. cluster size</td>
<td>1844</td>
<td>3713</td>
</tr>
<tr>
<td>Min. cluster size</td>
<td>52</td>
<td>18</td>
</tr>
<tr>
<td>Med. cluster size</td>
<td>131</td>
<td>45</td>
</tr>
<tr>
<td>Avg. cluster size</td>
<td>313</td>
<td>269</td>
</tr>
</tbody>
</table>


\(^4\)Reuters-21578 corpus is at http://www.daviddlewis.com/resources/testcollections/reuters21578/
map provides the statistics of the two document corpora. 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 (MTI) are used to measure the clustering performance [21][6]. Given a document \( x \), 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. 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) [22], and Normalized Cut (NC in short) [18][15]. Spectral clustering methods have recently emerged as one of the most effective document clustering

<table>
<thead>
<tr>
<th>( k )</th>
<th>PLSI</th>
<th>LDA</th>
<th>LapPLSI</th>
<th>( k )-means</th>
<th>AA</th>
<th>NC</th>
<th>NMF</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>91.5</td>
<td>93.1</td>
<td>99.8</td>
<td>97.6</td>
<td>99.8</td>
<td>97.7</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>82.3</td>
<td>88.2</td>
<td>99.7</td>
<td>90.4</td>
<td>90.6</td>
<td>97.9</td>
<td>95.9</td>
</tr>
<tr>
<td>4</td>
<td>76.0</td>
<td>80.8</td>
<td>99.3</td>
<td>86.3</td>
<td>86.4</td>
<td>95.9</td>
<td>93.2</td>
</tr>
<tr>
<td>5</td>
<td>70.0</td>
<td>77.6</td>
<td>98.8</td>
<td>81.1</td>
<td>81.6</td>
<td>94.7</td>
<td>89.9</td>
</tr>
<tr>
<td>6</td>
<td>69.0</td>
<td>73.6</td>
<td>97.5</td>
<td>79.2</td>
<td>79.4</td>
<td>93.4</td>
<td>91.4</td>
</tr>
<tr>
<td>7</td>
<td>63.7</td>
<td>67.4</td>
<td>96.3</td>
<td>73.8</td>
<td>80.4</td>
<td>89.1</td>
<td>85.8</td>
</tr>
<tr>
<td>8</td>
<td>59.8</td>
<td>65.1</td>
<td>94.1</td>
<td>72.5</td>
<td>73.8</td>
<td>85.0</td>
<td>82.3</td>
</tr>
<tr>
<td>9</td>
<td>63.2</td>
<td>66.5</td>
<td>93.8</td>
<td>73.6</td>
<td>73.6</td>
<td>86.0</td>
<td>83.9</td>
</tr>
<tr>
<td>10</td>
<td>60.7</td>
<td>65.8</td>
<td>92.8</td>
<td>72.3</td>
<td>73.5</td>
<td>81.4</td>
<td>82.6</td>
</tr>
<tr>
<td>Avg</td>
<td>70.7</td>
<td>75.3</td>
<td>96.9</td>
<td>80.8</td>
<td>81.5</td>
<td>91.5</td>
<td>89.4</td>
</tr>
</tbody>
</table>

Where \( k \) is the number of clusters

Table 5: Clustering performance on TDT2

![Figure 1](image)

Figure 1: (a) Accuracy vs. the number of classes on TDT2 corpus (b) Normalized mutual information vs. the number of classes on TDT2 corpus
averaging the scores from the 50 tests. 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 edge is assigned a weight to reflect the similarity between 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 cut criteria they used. Interestingly, Zha et al. [22] 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 in short) [21], which has been shown to be a very effective document 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 LapLSI approach gets significantly better performance than PLSI and LDA. Moreover, LapLSI 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, LapLSI can have better hidden topic modeling power in the sense of semantic structure.

- The improvement of LapLSI 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.

4.2.4 Parameter Selection

Our LapLSI model has two essential parameters: the number

<table>
<thead>
<tr>
<th>$k$</th>
<th>PLSI</th>
<th>LDA</th>
<th>LapPLSI</th>
<th>$k$-means</th>
<th>AA</th>
<th>NC</th>
<th>NMF</th>
<th>PLSI</th>
<th>LDA</th>
<th>LapPLSI</th>
<th>$k$-means</th>
<th>AA</th>
<th>NC</th>
<th>NMF</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>72.6</td>
<td>79.1</td>
<td>93.1</td>
<td>82.2</td>
<td>86.3</td>
<td>87.0</td>
<td></td>
<td>23.4</td>
<td>39.1</td>
<td>69.1</td>
<td>39.7</td>
<td>45.0</td>
<td>56.3</td>
<td>55.7</td>
</tr>
<tr>
<td>3</td>
<td>65.8</td>
<td>69.2</td>
<td>88.4</td>
<td>68.7</td>
<td>73.0</td>
<td>78.7</td>
<td>77.6</td>
<td>29.2</td>
<td>39.5</td>
<td>65.5</td>
<td>44.5</td>
<td>43.6</td>
<td>54.2</td>
<td>54.3</td>
</tr>
<tr>
<td>4</td>
<td>56.7</td>
<td>59.6</td>
<td>79.9</td>
<td>62.2</td>
<td>63.1</td>
<td>74.5</td>
<td>74.5</td>
<td>31.4</td>
<td>38.8</td>
<td>59.5</td>
<td>47.3</td>
<td>43.5</td>
<td>52.5</td>
<td>56.2</td>
</tr>
<tr>
<td>5</td>
<td>52.1</td>
<td>56.5</td>
<td>75.2</td>
<td>59.6</td>
<td>59.2</td>
<td>72.0</td>
<td>71.4</td>
<td>35.1</td>
<td>42.3</td>
<td>60.5</td>
<td>51.1</td>
<td>48.5</td>
<td>56.7</td>
<td>58.6</td>
</tr>
<tr>
<td>6</td>
<td>52.6</td>
<td>53.2</td>
<td>72.5</td>
<td>59.7</td>
<td>58.6</td>
<td>70.2</td>
<td>68.7</td>
<td>37.3</td>
<td>43.9</td>
<td>59.4</td>
<td>54.3</td>
<td>50.7</td>
<td>56.4</td>
<td>59.2</td>
</tr>
<tr>
<td>7</td>
<td>45.5</td>
<td>47.3</td>
<td>71.8</td>
<td>53.8</td>
<td>54.0</td>
<td>64.1</td>
<td>63.6</td>
<td>34.6</td>
<td>39.2</td>
<td>58.6</td>
<td>49.7</td>
<td>45.1</td>
<td>50.8</td>
<td>53.4</td>
</tr>
<tr>
<td>8</td>
<td>45.6</td>
<td>46.8</td>
<td>66.8</td>
<td>50.2</td>
<td>47.6</td>
<td>59.9</td>
<td>54.1</td>
<td>34.2</td>
<td>37.2</td>
<td>52.1</td>
<td>47.2</td>
<td>42.7</td>
<td>47.6</td>
<td>46.5</td>
</tr>
<tr>
<td>9</td>
<td>41.1</td>
<td>42.2</td>
<td>62.6</td>
<td>44.5</td>
<td>43.2</td>
<td>57.6</td>
<td>52.8</td>
<td>31.1</td>
<td>34.6</td>
<td>46.7</td>
<td>42.5</td>
<td>37.7</td>
<td>44.1</td>
<td>45.1</td>
</tr>
<tr>
<td>10</td>
<td>43.0</td>
<td>44.5</td>
<td>61.0</td>
<td>47.0</td>
<td>44.2</td>
<td>57.0</td>
<td>53.3</td>
<td>35.1</td>
<td>38.5</td>
<td>51.5</td>
<td>47.4</td>
<td>42.6</td>
<td>48.3</td>
<td>49.1</td>
</tr>
</tbody>
</table>

**Avg** | 52.8 | 55.4| 74.6   | 58.3       | 58.3| 68.9| 67.0| 32.4 | 39.2| 58.1   | 47.1       | 44.4| 51.9| 53.1|
of nearest neighbors $p$ and the regularization parameter $\lambda$. Figure 3 and Figure 4 show how the performance of LapPLSI varies with the parameters $\lambda$ and $p$, respectively. As we can see, the LapPLSI is very stable with respect to both the parameter $\lambda$ and $p$. It achieves consistent good performance with the $\lambda$ varying from 500 to 50000 and $p$ varying from 5 to 11.

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 $\lambda$ which controls the smoothness of our LapPLSI model. LapPLSI boils down to original PLSI when $\lambda = 0$. Also, it is easy to see that $P(z_k|d_i)$ will be the same for all the documents when $\lambda = +\infty$. Thus, a suitable value of $\lambda$ 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.

Acknowledgment

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.
Figure 4: The performance of LapPLSI vs. parameter $p$. LapPLSI achieves good performance with the parameter $p$ varying from 5 to 11.

6. REFERENCES


