Probabilistic Latent Semantic Indexing (PLSI)

CS 290D Paper Presentation

Presenter: Sina Miran
Outline

1. Introduction
2. Latent Semantic Indexing (LSI)
3. PLSI Model Definition
4. Fitting the Model on the Data
5. Example Output
6. Final Remark
7. References
As more information becomes available, it becomes more difficult to find and discover what we need.

We need tools to help us organize, search and understand these vast amount of information.

Topic modeling provides methods for automatically organizing, understanding, searching, and summarizing large electronic archives:

1. Discover the hidden themes in the collection
2. Annotate the documents according to these themes
3. Use annotations to organize, summarize and search
1. Introduction

Today, the large collection of data calls for **unsupervised** probabilistic models.

Example Applications:

1. Summarizing Collections of Images
   - Sky Water Tree
   - Scotland Water
   - Flower Hills Tree

2. Evolution of Pervasive Topics by Time

---

**Probabilistic Latent Sematic Indexing (PLSI)**
2. Latent Semantic Indexing (LSI)

Some Assumptions:

• We have a set of documents $d_1, d_2, \ldots, d_N$.

• Each document is just a collection of words or a “bag of words”. Thus, the order of the words and the grammatical role of the words (subject, object, verbs, ...) are not considered in the model.

• Words like am/is/are/of/a/the/but/... (stop words) can be eliminated from the documents as a preprocessing step since they don’t carry any information about the “topics”.

• In fact, we can eliminate words that occur in at least %80 ~ %90 of the documents!
2. Latent Semantic Indexing (LSI)

Document-Term Matrix:

Each row represents a document
Each column includes the count of the corresponding term in each of the documents

\[
\begin{pmatrix}
D_1 & w_{11} & w_{12} & \ldots & w_{1t} \\
D_2 & w_{21} & w_{22} & \ldots & w_{2t} \\
: & : & : & \ldots & : \\
: & : & : & \ldots & : \\
D_n & w_{n1} & w_{n2} & \ldots & w_{nt}
\end{pmatrix}
\]

<table>
<thead>
<tr>
<th>Document</th>
<th>Above</th>
<th>Attempt</th>
<th>Absent</th>
<th>Absolute</th>
<th>Absorb</th>
<th>Accurate</th>
<th>Acceptable</th>
<th>Access</th>
</tr>
</thead>
<tbody>
<tr>
<td>Beige Book - National Summary - 1970-05-20</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>Beige Book - National Summary - 1970-06-17</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>Beige Book - National Summary - 1970-07-15</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>Beige Book - National Summary - 1970-08-12</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>Beige Book - National Summary - 1970-09-09</td>
<td>0</td>
<td>0</td>
<td>2</td>
<td>0</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>Beige Book - National Summary - 1970-10-14</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>Beige Book - National Summary - 1970-11-11</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>Beige Book - National Summary - 1970-12-09</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
</tbody>
</table>
Latent Semantic Indexing (LSI)

**LSI**: Perform a low-rank approximation of document-term matrix (typical rank 100-300)

**General Idea:**
- Map documents (and terms) to a low-dimensional representation (PCA).
- Design a mapping such that the low-dimensional space reflects **semantic associations** (latent semantic space).
- Compute document similarity based on the **inner product** in the **latent semantic space**.

**Goals:**
- Similar terms and documents map to similar location in low-dimensional space
2. Latent Semantic Indexing (LSI)

Singular Value Decomposition (SVD) Review:

\[ A = U \Sigma V' \in \mathbb{R}^{n \times m} \]

\[ U \in \mathbb{R}^{n \times k} \quad \Sigma \in \mathbb{R}^{k \times k} \quad V \in \mathbb{R}^{m \times k} \]

\[ U'U = I \quad V'V = I \quad \Sigma = \text{diag}(\sigma_1, \ldots, \sigma_k), \sigma_i \geq \sigma_{i+1} \quad k = \text{rank}(A) \]

Approximation Problem:

\[ X^* = \arg\min_{\hat{X} : \text{rank}(\hat{X}) = q} \|X - \hat{X}\|_F, \quad \text{Ferbenius Norm} \|A\|_F \overset{\text{def}}{=} \sqrt{\sum \sum |a_{ij}|^2} \]
Latent Semantic Indexing (LSI)

\[ X^* = U \text{diag}(\sigma_1, \ldots, \sigma_q, 0, \ldots, 0)V' \]

\[ X^* = \sum_{r=1}^{q} \sigma_r u_r v'_r \]

Similarity measure: inner products
Vocabulary Mismatch Problem:
- One concept can be represented by several different words!
- Two documents might not contain similar terms (for instance due to writing styles) but refer to a single concept.
- Queries can contain words not present in a document and still be very relevant to that document!

We’re somehow looking for \( P(\text{a word or a query | the context}) \):

\[
P(R_d = 1|q) = \frac{P(q|R_d = 1)P(R_d = 1)}{P(q)} \propto \frac{P(q|R_d = 1)}{P(R_d = 1)}
\]

- \( R_d \in \{0,1\} \): relevance of a document
- \( q \): a query, set of words
- \( P(R_d = 1) \): Uniform or relevant to the popularity of the document
- \( P(q|R_d = 1) \): Given a document how probable is a query
- \( P(q) \): the context
3. PLSI Model Definition

How to calculate $P(q|R_d = 1)$:

- For each document calculate the probability of each word $w$ coming from (or being relevant to) that document i.e. $P(w|R_d = 1)$
- Calculate the conditional probability of the words in $q$

PLSI Model Elements:

- A set of documents $\{d_1, \ldots, d_N\}$
- A set of concepts, classes or topics $\{z_1, \ldots, z_K\}$
- A set of words $\{w_1, \ldots, w_M\}$

**Problem:** We only have the realizations of the words, and concepts are not observed (they are latent). How can we infer the probabilities of different words and concepts from the at hand documents???
The generative process:

- Each **concept** is a distribution over words.
- Each **document** is a mixture of corpus-wide topics.
- Each **word** is drawn from one of these topics.
- We only observe the words within the documents and the other structures are hidden variables.
3. PLSI Model Definition

- Select a document with probability $P(d)$
- Pick a latent class $z$ with probability $P(z|d; \theta)$
- Generate a word $w$ with probability $P(w|z; \pi)$

\[
P(d, w) = P(d)P(w|d)
\]

\[
\hat{P}_{LSA}(w|d) = \sum_{z \in Z} P(w|z; \theta)P(z|d; \pi)
\]

\[
\hat{P}_{LSA}(d, w) = P(d) \sum_{z \in Z} P(w|z)P(z|d) = \sum_{z \in Z} P(d|z)P(z)P(w|z)
\]
This can be demonstrated as a matrix factorization

\[ \hat{P}_{LSA}(d, w) = \sum_{z \in Z} P(d|z)P(z)P(w|z) \]

Contrast to SVD:
- No orthonormality condition for \( U \) and \( V \) here.
- The elements of \( U \) and \( V \) are non-negative.
Maximum Likelihood Estimation (ML):

Find all the parameters such that the probability of observing the corpus is maximized.

Likelihood function to be maximized: \( L = \prod_{i=1}^{N} \prod_{j=1}^{M} P(d_i, w_j)^{n(d_i, w_j)} \)

\[
\log L = \sum_{i=1}^{N} \sum_{j=1}^{M} n(d_i, w_j) \log P(d_i, w_j) = \sum_{i=1}^{N} \sum_{j=1}^{M} n(d_i, w_j) \log \sum_{k=1}^{K} P(d_i)P(z_k | d_i)P(w_j | z_k) \\
= \sum_{i=1}^{N} n(d_i) \left[ \log P(d_i) + \sum_{j=1}^{M} \frac{n(d_i, w_j)}{n(d_i)} \log \left[ \sum_{k=1}^{K} P(z_k | d_i)P(w_j | z_k) \right] \right]
\]
4. Fitting the Model on the Data

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

Estimated directly from data:
- \( P(d_i) \): uniform or related to popularity of the document \( d_i \)
- \( n(d_i) \): number of words in \( d_i \)
- \( n(d_i, w_j) \): count of word \( w_j \) in \( d_i \)

The coupling effects of \( z_k \) makes this a hard optimization problem!

We use the standard Expectation Maximization (EM) algorithm to find an optimal solution.
4. Fitting the Model on the Data

**EM Algorithm:**

Finding the solution of ML or MAP when some data is missing i.e. we have latent variables not observed.

Given: observations $c = (c_1, ..., c_n)$ of the random variable $X$.

Model: $(X, Z) \sim p_{\theta}$ for some unknown parameter $\theta$.

Goal: $\theta_{ML} = \arg\max_{\theta} p_{\theta}(x)$ since we only observe $X$

Issue: we have a marginal probability $p_{\theta}(x) = \sum_z p_{\theta}(x, z)$ which is difficult to maximize analytically (mainly because of the sum)

Also, you will probably have local maxima!
EM Algorithm improves \( p_\theta(x) \) in two iterative steps namely E step and M step such that \( p_{\theta_{t+1}}(x) \geq p_{\theta_t}(x) \)

**Note:** Since you may have local maxima, EM might not give you the global optimum i.e. \( \theta_{ML} \)

Back to our problem:

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

EM for our problem (repeat until convergence):

1. **E-step:** Calculate posterior probabilities for latent variables given the observations and current estimates
2. **M-step:** Update parameters using the posterior probabilities in E-step to increase \( \log L \)
4. Fitting the Model on the Data

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

1. **E-step:** Calculating posterior probabilities using the current estimates

\[ P(z_k | d_i, w_j) = \frac{P(w_j, z_k | d_i)}{P(w_j | d_i)} = \frac{P(w_j | z_k, d_i)P(z_k | d_i)}{\sum_{i=1}^{K} P(w_j | z_i, d_i)P(z_i | d_i)} \]

2. **M-step:** Maximizing \( \log L \) having the posterior probability

\[ P(w_j | z_k) = \frac{\sum_{i=1}^{N} n(d_i, w_j)P(z_k | d_i, w_j)}{\sum_{m=1}^{M} \sum_{i=1}^{N} n(d_i, w_m)P(z_k | d_i, w_m)} \quad \quad P(z_k | d_i) = \frac{\sum_{j=1}^{M} n(d_i, w_j)P(z_k | d_i, w_j)}{n(d_i)} \]
Data Set: Topic Detection and Tracking corpus (TDT-1)
- Approximately 7 million words
- 15863 documents
- $K=128$

Two most probable topics that generate The terms “flight” and “love.

<table>
<thead>
<tr>
<th>“plane”</th>
<th>“space shuttle”</th>
<th>“family”</th>
<th>“Hollywood”</th>
</tr>
</thead>
<tbody>
<tr>
<td>plane</td>
<td>space</td>
<td>home</td>
<td>film</td>
</tr>
<tr>
<td>airport</td>
<td>shuttle</td>
<td>family</td>
<td>movie</td>
</tr>
<tr>
<td>crash</td>
<td>mission</td>
<td>like</td>
<td>music</td>
</tr>
<tr>
<td>flight</td>
<td>astronauts</td>
<td>love</td>
<td>new</td>
</tr>
<tr>
<td>safety</td>
<td>launch</td>
<td>kids</td>
<td>best</td>
</tr>
<tr>
<td>aircraft</td>
<td>station</td>
<td>mother</td>
<td>hollywood</td>
</tr>
<tr>
<td>air</td>
<td>crew</td>
<td>life</td>
<td>love</td>
</tr>
<tr>
<td>passenger</td>
<td>nasa</td>
<td>happy</td>
<td>actor</td>
</tr>
<tr>
<td>board</td>
<td>satellite</td>
<td>friends</td>
<td>entertainment</td>
</tr>
<tr>
<td>airline</td>
<td>earth</td>
<td>cnn</td>
<td>star</td>
</tr>
</tbody>
</table>
5. Example Output

Data Set: Topic Detection and Tracking corpus (TDT-1)
- Approximately 7 million words
- 15863 documents
- K=128

Four additional topics from the 128 topic-Decomposition of the TDT-1 corpus.

<table>
<thead>
<tr>
<th>“Bosnia”</th>
<th>“Iraq”</th>
<th>“Rwanda”</th>
<th>“Kobe”</th>
</tr>
</thead>
<tbody>
<tr>
<td>un</td>
<td>iraq</td>
<td>refugees</td>
<td>building</td>
</tr>
<tr>
<td>bosnian</td>
<td>iraqi</td>
<td>aid</td>
<td>city</td>
</tr>
<tr>
<td>serbs</td>
<td>sanctions</td>
<td>rwanda</td>
<td>people</td>
</tr>
<tr>
<td>bosnia</td>
<td>kuwait</td>
<td>relief</td>
<td>rescue</td>
</tr>
<tr>
<td>serb</td>
<td>un</td>
<td>people</td>
<td>buildings</td>
</tr>
<tr>
<td>sarajevo</td>
<td>council</td>
<td>camps</td>
<td>workers</td>
</tr>
<tr>
<td>nato</td>
<td>gulf</td>
<td>zaire</td>
<td>kobe</td>
</tr>
<tr>
<td>peacekeepers</td>
<td>saddam</td>
<td>camp</td>
<td>victims</td>
</tr>
<tr>
<td>nations</td>
<td>baghdad</td>
<td>food</td>
<td>area</td>
</tr>
<tr>
<td>peace</td>
<td>hussein</td>
<td>rwandan</td>
<td>earthquake</td>
</tr>
</tbody>
</table>
Data Set: CLUSTER generated by the author
- Abstracts of 1568 documents on clustering
- K=128

Eight selected topics from the 128 topic- Decomposition of CLUSTER

<table>
<thead>
<tr>
<th>“segment 1”</th>
<th>“segment 2”</th>
<th>“matrix 1”</th>
<th>“matrix 2”</th>
<th>“line 1”</th>
<th>“line 2”</th>
<th>“power 1”</th>
<th>power 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>imag SEGMENT</td>
<td>texture</td>
<td>color</td>
<td>tissue</td>
<td>brain</td>
<td>slice</td>
<td>cluster</td>
<td>mri</td>
</tr>
<tr>
<td></td>
<td>segmentind.</td>
<td>speaker</td>
<td>speech</td>
<td>recogni</td>
<td>signal</td>
<td>train</td>
<td>hmm</td>
</tr>
<tr>
<td></td>
<td></td>
<td>robust</td>
<td>MATRIX</td>
<td>eigenvalu</td>
<td>uncertai</td>
<td>plane</td>
<td>line</td>
</tr>
<tr>
<td></td>
<td></td>
<td>linear</td>
<td>MATRIX</td>
<td>cellular</td>
<td>design</td>
<td>machinepart</td>
<td>format</td>
</tr>
<tr>
<td></td>
<td></td>
<td>condition</td>
<td>perturb</td>
<td>root</td>
<td>suffici</td>
<td>segunent</td>
<td>fundamen</td>
</tr>
<tr>
<td></td>
<td></td>
<td>segmentind.</td>
<td>format</td>
<td>group</td>
<td></td>
<td>recogn</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>impos</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Probabilistic Latent Semantic Indexing (PLSI)
We want $P(z|d_i)$ to be sparse over $z_1, ..., z_k$ i.e. each document should be related to a small number of topics.

Also, we want $P(w|z_k)$ to be sparse over $w_1, ..., w_M$ i.e. each topic should be associated with a small proportion of the words.

Note that no conditions have been enforced on $p(z|d)$ and $p(w|z)$ in the PLSI model.

Enforcing the sparsity conditions with a Dirichlet distribution on $p(z|d)$ and $p(w|z)$ leads to the Latent Dirichlet Allocation (LDA) Model which will be presented next.
7. References


[2] Video Lectures of Thomas Hofmann and David Blei on videolectures.net:
http://videolectures.net/slsfs05_hofmann_lsvm/
http://videolectures.net/mlsso9uk_blei_tm/

[3] Homepage of Thomas Hofmann, Assistant Professor of CS at Brown University:
http://cs.brown.edu/~th/

[4] Homepage of David Blei, Associate Professor of CS at Princeton University:
Questions?!