Probabilistic
Latent Semantic Indexing

Thomas Hofmann
Proceedings of the Twenty-Second Annual International SIGIR
Conference on Research and Development in Information Retrieval
(1999)

Presentation by Jonathan Ultis
Computer Science and Engineering Department
University of California, San Diego
jultis@cs.ucsd.edu

Outline

I. Latent Semantic Analysis
II. Singular Value Decomposition
III. Probabilistic Latent Semantic Analysis
IV. Experiments
V. The PLSA Advantage
Issues in Information Retrieval

- Synonyms are separate words that have the same meaning. They tend to reduce recall.
- Polysemy refers to words that have multiple meanings. This problem tends to reduce precision.
- Both issues point to a more general problem. There is a disconnect between topics and keywords.

Latent Semantic Analysis (LSA)

- LSA attempts to discover information about the meaning behind words.
- Highly correlated words usually indicate the existence of a topic.
- LSA is proposed as an automated solution to the problems of synonymy and polysemy.
Indexing by Latent Semantic Analysis

- Once LSA has been performed, documents can be written as vectors in latent semantic space rather than as word vectors. This is known as Latent Semantic Indexing (LSI).
- This paper proposed Singular Value Decomposition (SVD) as an appropriate technique for LSA.
- SVD is still the most commonly used technique for LSA.


SVD Overview

- Documents are commonly represented as vectors of term frequencies.
- Multiple documents can be represented as a matrix $X$ of terms by documents.
SVD is a standard technique for breaking a matrix into orthogonal components.

- $m$ is the rank of the original matrix. $m$ is usually much smaller than either $t$ or $d$.
- The matrix $S_0$ is a diagonal matrix of singular values.
- $T_0$ and $D_0$ are matrices of orthonormal columns and rows respectively.

The largest entries in $S_0$ represent the dimensions of greatest variance. These represent strong divisions in term usage.

- LSA is performed by removing small entries in $S_0$.
- $X$ will be as close to $X_0$ as possible for a rank $k$ matrix (least-squares-fit).
Least Squares Fit

- The least-squares-fit error has the minimum sum squared error for a matrix of its rank.
- Example:
  Given the vector \{1, 5, 1\}
  The vector \{2, 5, 2\} has a sum squared error of 2.
  The vector \{1, 8, 1\} has a sum squared error of 9.
  Therefore \{1, 5, 1\} is closer to \{2, 5, 2\} in a least-squares sense.

Probabilistic Latent Semantic Analysis

- PLSA is based on a generative probabilistic model.
- Documents generate a particular distribution of aspects (topics).
- Aspects generate a particular distribution of word usage.
The probability of each document and the probability of each word are known.

The probability of an aspect given a document is unknown.

The probability of a word given an aspect is unknown.

Visualizations of equation 2

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

- Maximum Likelihood

- The EM algorithm is used to estimate the unknowns by maximizing the log likelihood of the training data.

\[
\mathcal{L} = \sum_{d \in D} \sum_{w \in W} n(d, w) \log P(d, w)
\]

- Example:

  Given the vector \(\{1, 5, 1\}\) and a perfectly trained model of one document \(P(w) = \{0.14, 0.71, 0.14\}\)

  The vector \(\{2, 5, 2\}\) has a likelihood of \(-4.16\).

  The vector \(\{1, 8, 1\}\) has a likelihood of \(-2.90\).

  Therefore \(\{1, 5, 1\}\) is closer to \(\{1, 8, 1\}\) in a log likelihood sense.
The EM algorithm performs an implicit clustering of words into aspects.

<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>

Each column holds the 10 words that a particular aspect is most likely to generate. Column headings were assigned by a human.

- The EM algorithm performs an implicit clustering of words into aspects.

Effectiveness of PLSA

- The effectiveness of PLSA is tested by using the aspects to perform queries (PLSI).
- Two basic query techniques for PLSI are tested against the normal query technique for LSI.
- All queries use the cosine similarity metric to find the similarity between vectors.
PLSI-U incorporates TF-IDF weighting by directly modifying the word vectors of the document and query before performing the comparison.

PLSI-Q uses a weighted average of the TF-IDF scores of the words that affect each aspect.

The aspect vector for a query is generated by treating the query as a new document. The query is added to the model and the weights for the query are trained with the EM algorithm.
PLSI-Q* and PLSI-U*

- The two basic query techniques are also tested using combinations of models.
- PLSI-U* combines the augmented word vectors predicted by models with different numbers of aspects.
- PLSI-Q* combines the cosine similarities of the aspect vectors predicted by models with different numbers of aspects.

Experiments

- 4 document collections provide a test bed.
  - MED – 1033 abstracts from the National Library of Medicine
  - CRAN – 1400 documents on aeronautics
  - CACM – 3204 abstracts from the CACM journals
  - CISI – 1460 abstracts in library science
- Average precision across 9 recall levels is reported (10, 20, …, 90%).
- Average relative improvement over the baseline at each recall level is reported.
PLSI-U is better than LSI in every case.
PLSI-U* is the champion in almost every case.
The author speculates that PLSI-Q* could do much better if tf-idf weighting were incorporated more effectively.
The PLSA Advantage

- PLSA has several advantages over traditional SVD based LSA.
- PLSA attempts to maximize the likelihood of the data rather than minimizing the sum squared error.
- PLSA can use the usual methods to prevent overfitting, which can lead to more general models.
• Models can be combined productively with PLSA.
• PLSA provides a “more intuitive” definition for aspects.
• Empirical results bear out these advantages.