Probabilistic Latent Semantic Analysis

Yuriy Sverchkov

Intelligent Systems Program
University of Pittsburgh

October 6, 2011

Outline

Latent Semantic Analysis (LSA)
- A quick review

Probabilistic LSA (pLSA)
- The pLSA model

Learning
- EM and tempered EM

Applications
- pLSI and pHITS

Outline

Latent Semantic Analysis (LSA)

A quick review

Probabilistic LSA (pLSA)

The pLSA model

Learning

EM and tempered EM

Applications

pLSI and pHITS

LSA: A quick review

LSA uses PCA to find a lower-dimensional "topic" space.

$$\begin{array}{cc} & \text{documents} \\ \text{terms} & \begin{pmatrix} x_{1,1} & \cdots & x_{1,n} \\ \vdots & \ddots & \vdots \\ x_{m,1} & \cdots & x_{m,n} \end{pmatrix} \end{array} =$$

$$\begin{array}{cc} & \text{topics} \\ \text{terms} & \begin{pmatrix} w_{1,1} & \cdots & w_{1,r} \\ \vdots & \ddots & \vdots \\ w_{m,1} & \cdots & w_{m,r} \end{pmatrix} \end{array} \begin{pmatrix} \text{topic weights} & \cdots & 0 \\ \vdots & & \vdots \\ 0 & \cdots & \end{pmatrix} \begin{array}{cc} & \text{documents} \\ \text{topics} & \begin{pmatrix} v_{1,1} & \cdots & v_{1,n} \\ \vdots & \ddots & \vdots \\ v_{r,1} & \cdots & v_{r,n} \end{pmatrix} \end{array}$$

TERMS $\longleftrightarrow$ **TOPICS** $\longleftrightarrow$ **DOCUMENTS**

PCA as reconstruction error minimization

For each data vector $\mathbf{x}_n = (x_{n1}, \ldots, x_{nd})$, and for $M < d$, find $U = (\mathbf{u}_1, \ldots, \mathbf{u}_M)$ that minimizes

$$E_M \equiv \sum_{n=1}^{N} \|\mathbf{x}_n - \hat{\mathbf{x}}_n\|^2$$

where $\hat{\mathbf{x}}_n = \bar{\mathbf{x}} + \sum_{i=1}^{M} y_{ni}\mathbf{u}_i$ and $\bar{\mathbf{x}} \equiv \frac{1}{N}\sum_{n=1}^{N}\mathbf{x}_n$ giving:

$$E_M = \sum_{i=M+1}^{d} \sum_{n=1}^{N} [\mathbf{u}_i^T(\mathbf{x}_n - \hat{\mathbf{x}})]^2 = \sum_{i=M+1}^{d} \mathbf{u}_i^T \Sigma \mathbf{u}_i = \sum_{i=M+1}^{d} \lambda_i$$

Outline

Latent Semantic Analysis (LSA)
- A quick review

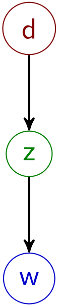
Probabilistic LSA (pLSA)
- The pLSA model

Learning
- EM and tempered EM

Applications
- pLSI and pHITS

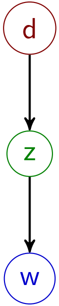
Probabilistic LSA

- The same "document $\leftrightarrow$ topic $\leftrightarrow$ word" idea in a probabilistic framework.
- Asymmetric generative aspect model:
  1. Select a document d with probability $P(d)$.
  2. Select a latent class z with probability $P(z|d)$.
  3. Generate a word w with probability $P(w|z)$.
- A *mixture model*
  - Each document corresponds to a mixture of topics.
  - Each topic corresponds to a mixture of words.



Parametrization

- d The index of a document in the dataset.
- $P(d)$ The frequency of the document in the corpus (uniform in practice).
- z The index of a topic.
- $P(z|d)$ Latent parameters that define the distribution of topics for a particular document.
- w The index of a word in the dictionary.
- $P(w|z)$ Latent parameters that define the distribution of words for a particular topics.



Independence

- Remember independence equivalence classes in Bayesian networks?

$A \perp C | B$

$$A \rightarrow B \rightarrow C$$

$$A \leftarrow B \leftarrow C$$

$$A \leftarrow B \rightarrow C$$

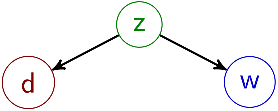
$A \perp C$

$$A \rightarrow B \leftarrow C$$

Symmetric aspect model

Parametrization

$$P(d, w) = \sum_{z \in \mathcal{Z}} P(z)P(d|z)P(w|z)$$



- Inference is BN inference.
- Learning is the same as for any BN with latent variables: EM.

pLSA vs LSA

pLSA

- Assumes conditional independence given a lower-dimensional variable.
- Maximizes likelihood function.
- Parameters are multinomial distributions.
- EM is slow.
- EM converges to a local optimum.

LSA

- Assumes linear transformation to a low-dimensional space.
- Minimizes Gaussian error.
- Parameters have no obvious interpretation.
- Linear operations are fast.
- SVD is exact (up to numerical precision).

Outline

Latent Semantic Analysis (LSA)

A quick review

Probabilistic LSA (pLSA)

The pLSA model

Learning

EM and tempered EM

Applications

pLSI and pHITS

Learning: standard EM

- E-step:

$$P(z|d,w) = \frac{P(z)P(d|z)P(w|z)}{\sum_{z'\in\mathcal{Z}} P(z')P(d|z')P(w|z')}$$

- M-step:

$$P(w|z) \propto \sum_{d\in\mathcal{D}} n(d,w)P(z|d,w)$$

$$P(d|z) \propto \sum_{w\in\mathcal{W}} n(d,w)P(z|d,w)$$

$$P(z) \propto \sum_{d\in\mathcal{D}} \sum_{w\in\mathcal{W}} n(d,w)P(z|d,w)$$

Learning: tempered EM (TEM)

- New E-Step:

$$P(z|d,w) = \frac{P(z)\left[P(d|z)P(w|z)\right]^{\beta}}{\sum_{z' \in \mathcal{Z}} P(z')\left[P(d|z')P(w|z')\right]^{\beta}}$$

- Same as the standard E-Step when $\beta = 1$.
- Same as a posterior given uniform data when $\beta = 0$.
- Algorithm:
    1. Hold out some data.
    2. Set $\beta \leftarrow 1$.
    3. Perform EM and decrease β at some rate ($\beta \leftarrow \eta\beta$ with $\eta < 1$).
    4. Stop if performance on held-out data doesn't increase, otherwise repeat previous step.
    5. Perform some final iterations on full data.

Outline

Latent Semantic Analysis (LSA)
- A quick review

Probabilistic LSA (pLSA)
- The pLSA model

Learning
- EM and tempered EM

Applications
- pLSI and pHITS

Applications to information retrieval and link analysis

- Information Retrieval: pLSI
  - Index documents by their topic (z) distributions.
  - Queries are computed by scoring each document with $P(w|d)$ (for words in the query).
  - Can fold-in a new query as a "hypothetical document" $P(z|q)$ by updating that probability with EM.
- Link analysis: pHITS
  - d are documents, c are citations (correspond to w in pLSA).
  - Want to group these into "communities" (z).
  - Authoritativeness measures:

$$
\begin{aligned}
P(c|z) &\quad \text{authority of } c \text{ within the community } z. \\
P(z|c) &\quad \text{topic-specific authority.} \\
P(z|c)P(c|z) &\quad \text{topic characteristic for community.}
\end{aligned}
$$