清華大學
Tsinghua University

第十七章
潜在语义分析

潜在语义分析

- 潜在语义分析（(latent semantic analysis, LSA）是一种无监督学习方法，主要用于文本的话题分析
- 通过矩阵分解发现文本与单词之间的基于话题的语义关系
- 文本信息处理中，传统的方法以单词向量表示文本的语义内容，以单词向量空间的度量表示文本之间的语义相似度。
- 潜在语义分析旨在解决这种方法不能准确表示语义的问题，试图从大量的文本数据中发现潜在的话题，以话题向量表示文本的语义内容，以话题向量空间的度量更准确地表示文本之间的语义相似度。这也是话题分析（topic modeling）的基本想法。

潜在语义分析

- 潜在语义分析使用的是非概率的话题分析模型。
- 具体地，将文本集合表示为单词-文本矩阵，对单词-文本矩阵进行奇异值分解，从而得到话题向量空间，以及文 在话题向量空间的表示。
- 奇异值分解特点是分解的矩阵正交
- 非负矩阵分解（non-negative matrix factorization, NMF）是另一种矩阵的因子分解方法，其特点是分解的矩阵非负
- 非负矩阵分解也可以用于话题分析

单词向量空间

- 文本信息处理，比如文本信息检索、文本数据挖掘的一个核心问题是对文本的语义内容进行表示，并进行文本之间的语义相似度计算。

- 最简单的方法是利用向量空间模型（vector space model, VSM），也就是单词向量空间模型（word vector space model)。

- 向量空间模型的基本想法是，给定一个文本，用一个向量表示该文本的“语义”
- 向量的每一维对应一个单词，其数值为该单词在该文本中出现的频数或权值
- 基 本假设是文本中所有单词的出现情况表示了文本的语义内容
- 文本集合中的每个文本都表示为一个向量，存在于一个向量空间
- 向量空间的度量，如内积或标准化内积表示文本之间的“语义相似度”。

单词向量空间

- 给定一个含有n个文本的集合 $D = \{d_1, d_2, \cdots, d_n\}$ ， 以及在所有文本中出现的m个单词的集合 $W = \{w_1, w_2, \cdots, w_m\}$。
- 将单词在文本中出现的 数据用一个单词-文本矩阵（word-document matrix）表示，记作X

$$X = \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1n} \\ x_{21} & x_{22} & \cdots & x_{2n} \\ \vdots & \vdots & & \vdots \\ x_{m1} & x_{m2} & \cdots & x_{mn} \end{bmatrix}$$

单词向量空间

- 这是一个 m x n 矩阵，元素 x_{ij} 表示单词 w_i 在文本.d_j 内中出现的频数或权值。

- 由于单 词的种类很多，而每个文本中出现单词的种类通常较少，所以单词-文本矩阵是一个稀疏矩阵。

单词向量空间

- 权值通常用单词频率-逆文本频率（term frequency-inverse document frequency, TF-IDF）表示，其定义是

$$\mathrm{TFIDF}_{ij} = \frac{\mathrm{tf}_{ij}}{\mathrm{tf}_{\bullet j}} \log \frac{\mathrm{df}}{\mathrm{df}_i}, \quad i = 1, 2, \cdots, m; \quad j = 1, 2, \cdots, n$$

- tf_{ij}：单词 w_i 出现在文本 d_j 中的频数
- $\mathrm{tf}_{\bullet j}$：是文本 d_j 中出现的所有单词的频 数之和
- df_i：含有单词 w_i 的文本数
- df：是文本集合D的全部文本数

单词向量空间

- 直观上，一个 单词在一个文本中出现的频数越高，这个单词在这个文本中的重要度就越高
- 一个单 词在整个文本集合中出现的文本数越少，这个单词就越能表示其所在文本的特点，重要度就越高
- 一个单词在一个文本的TF-IDF是两种重要度的积，表示综合重要度

单词向量空间

- 单词向量空间模型直接使用单词-文本矩阵的信息。单词-文本矩阵的第j列向量 x_j 表示文本 d_j

$$x_j = \begin{bmatrix} x_{1j} \\ x_{2j} \\ \vdots \\ x_{mj} \end{bmatrix}, \quad j = 1, 2, \cdots, n$$

- x_{ij} :单词 w_i 在文本 d_j 的权值
- 权值越大，该单词在该文本中的重要度就越高

$$X = [x_1 \quad x_2 \quad \cdots \quad x_n]$$

单词向量空间

- 两个单词向量的内积或标准化内积（余弦）表示对应的文本之间的语义相似度
- 因此，文本 d_i 与 d_j 之间的相似度为 $x_i \cdot x_j$， $\frac{x_i \cdot x_j}{\|x_i\|\|x_j\|}$
- 直观上，在两个文本中共同出现的单词越多，其语义内容就越相近，对应的单词向量同不为零的维度就越多，内积就越大（单词向量元素的值都是非负的），表示两个文本在语义内容上越相似

单词向量空间

- 单词向量空间模型
  - 模型简单
  - 计算效率高
  - 有局限性，内积相似度未必能够准确表达两个文本的语义相似度
    - 一词多义性(polysemy)
    - 多词一义性(synonymy)

例

- 单词向量空间模型中，文本 d_1 与 d_2 相似度并不高，尽管两个文本的内容相似，这是因为同义词“airplane”与“aircraft”被当作了两个独立的单词，单词向量空间模型不考虑单 的同义性，在此情况下无法进行准确的相似度计算。

	d_1	d_2	d_3	d_4
airplane	2			
aircraft		2		
computer			1	
apple			2	3
fruit				1
produce	1	2	2	1

例

- 文本 d_3 与 d_4 有一定的相似度，尽管两个文本的内容并不相似，这是因为单词“apple”具有多义，可以表示“apple computer”和“fruit",
  单词向量空间模型不考虑单词的多义性，
  在此情况下也无法进行准确的相似度计算。

	d_1	d_2	d_3	d_4
airplane	2			
aircraft		2		
computer			1	
apple			2	3
fruit				1
produce	1	2	2	1

话题向量空间

- 两个文本的语义相似度可以体现在两者的话题相似度上
- 一个文本一般含有若干个话题。 如果两个文本的话题相似，那么两者的语义应该也相似
- 话题可以由若干个语义相关的单词表示，同义词（如“airplane”与“aircraft"）可以表示同一个话题，而多义词（如“apple"）可以表示不同的话题。
- 这样，基于话题的模型就可以解决上述基于单 词的模型存在的问题。

话题向量空间

- 设想定义一种话题向量空间模型(topic vector space model)
- 给定一个文本，用话题空间的一个向量表示该文本，该向量的每一分量对应一个话题，其数值为该话题在该文本中出现的权值
- 用两个向量的内积或标准化内积表示对应的两个文本的语义相似度
- 注：单词向量空间模型与话题向量空间模型可以互为补充，现实中，两者可以同时使用。

话题向量空间

- 给定一个文本集合 $D=\{d_1, d_2, \cdots, d_n\}$ 和一个相应的单词集合 $W=\{w_1, w_2, \cdots, w_m\}$。可以获得其单词-文本矩阵X，X构成原始的单词向量空间，每一列是一个文本 在单词向量空间中的表示

$$X=\begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1n} \\ x_{21} & x_{22} & \cdots & x_{2n} \\ \vdots & \vdots & & \vdots \\ x_{m1} & x_{m2} & \cdots & x_{mn} \end{bmatrix}$$

- 矩阵X也可以写作 $X=[x_1 \quad x_2 \quad \cdots \quad x_n]$

话题向量空间

- 假设所有文本共含有k个话题。假设每个话题由一个定义在单词集合W上的m维向量表示，称为话题向量，即

$$t_l = \begin{bmatrix} t_{1l} \\ t_{2l} \\ \vdots \\ t_{ml} \end{bmatrix}, \quad l = 1, 2, \cdots, k$$

- t_{il}：单词 w_i 在话题 t_l 的权值，权值越大，该单词在该话题中的重要度就越高
- k个话题向量张成一个话题向量空间(topic vector space)，维数为k
- 话题向量空间T是单词向量空间X的一个子空间

话题向量空间

- 话题向量空间T也可以表示为一个矩阵，称为单词-话题矩阵（word-topic matrix），记作

$$T = \begin{bmatrix} t_{11} & t_{12} & \cdots & t_{1k} \\ t_{21} & t_{22} & \cdots & t_{2k} \\ \vdots & \vdots & & \vdots \\ t_{m1} & t_{m2} & \cdots & t_{mk} \end{bmatrix}$$

- 矩阵T也可写作

$$T = [t_1 \quad t_2 \quad \cdots \quad t_k]$$

文本在话题向量空间的表示

- 现在考虑文本集合D的文本 d_j，在单词向量空间中由一个向量 x_j 表示，将 x_j 投影到话题向量空间T中，得到在话题向量空间的一个向量y_j, y_j 是一个k维向量，其表达式为

$$y_j = \begin{bmatrix} y_{1j} \\ y_{2j} \\ \vdots \\ y_{kj} \end{bmatrix}, \quad j = 1, 2, \cdots, n$$

- y_{ij}：文本 d_j 在话题 t_i 的权值，权值越大，该话题在该文本中的重要度就越高

文本在话题向量空间的表示

- 矩阵Y表示话题在文本中出现的情况，称为话题-文本矩阵(topic-document matrix)，记作

$$Y = \begin{bmatrix} y_{11} & y_{12} & \cdots & y_{1n} \\ y_{21} & y_{22} & \cdots & y_{2n} \\ \vdots & \vdots & & \vdots \\ y_{k1} & y_{k2} & \cdots & y_{kn} \end{bmatrix}$$

- 矩阵Y可一个写作

$$Y = [y_1 \quad y_2 \quad \cdots \quad y_n]$$

从单词向量空间到话题向量空间的线性变换

- 这样一来，在单词向量空间的文本向量 x_j 可以通过它在话题空间中的向量 y_j 近似表示，具体地由k个话题向量以 y_j 为系数的线性组合近似表示

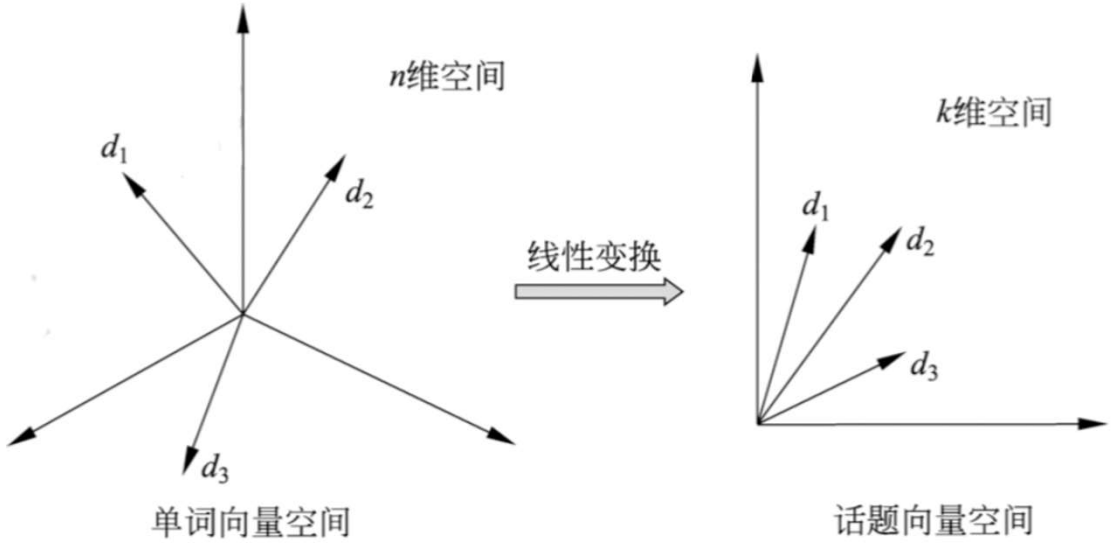
$$x_j \approx y_{1j}t_1 + y_{2j}t_2 + \cdots + y_{kj}t_k, \quad j = 1, 2, \cdots, n$$

- 所以，单词-文本矩阵X可以近似的表示为单词-话题矩阵T与话题—文本矩阵Y的乘积形式。这就是潜在语义分析。

$$X \approx TY$$

从单词向量空间到话题向量空间的线性变换

- 直观上，潜在语义分析是将文本在单词向量空间的表示通过线性变换转换为在话 题向量空间中的表示



从单词向量空间到话题向量空间的线性变换

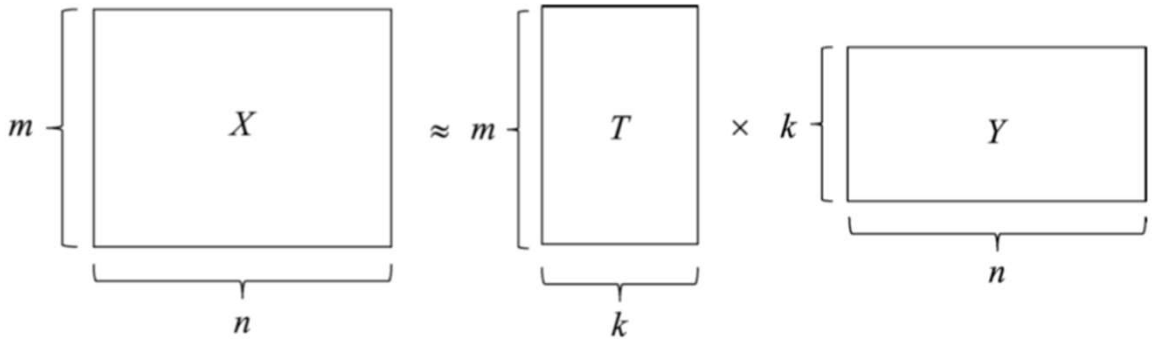


图 17.3　潜在语义分析通过矩阵因子分解实现，单词-文本矩阵 X 可以近似的表示为单词-话题矩阵 T 与话题-文本矩阵 Y 的乘积形式

从单词向量空间到话题向量空间的线性变换

- 在原始的单词向量空间中，两个文本 d_i 与 d_j 的相似度可以由对应的向量的内积 表示，即$x_i \cdot x_j$。
- 经过潜在语义分析之后，在话题向量空间中，两个文本 d_i 与 d_j 的相似度可以由对应的向量的内积即 $y_i \cdot y_j$ 表示。
- 要进行潜在语义分析，需要同时决定两部分的内容，一是话题向量空间T，二是文本在话题空间的表示Y，使两者的乘积是原始矩阵数据的近似，而这一结果完全从话题-文本矩阵的信息中获得

潜在语义分析算法

- 潜在语义分析利用矩阵奇异值分解

- 潜在语义对单词-文本矩阵进行奇异值分解，将其左矩阵作为话题向量空间，将其对角矩阵与右矩阵的乘积作为文本在话题向量空间的表示。

矩阵奇异值分解算法

- （1）单词-文本矩阵

- 给定文本集合 $D=\{d_1,d_2,\cdots,d_n\}$ 和单词集合 $W=\{w_1,w_2,\cdots,w_m\}$。潜在语
- 义分析首先将这些数据表成一个单词-文本矩阵

$$X=\begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1n} \\ x_{21} & x_{22} & \cdots & x_{2n} \\ \vdots & \vdots & & \vdots \\ x_{m1} & x_{m2} & \cdots & x_{mn} \end{bmatrix}$$

矩阵奇异值分解算法

- （2）截断奇异值分解

- 潜在语义分析根据确定的话题个数k对单词-文本矩阵X进行截断奇异值分解

$$X \approx U_k \Sigma_k V_k^{\mathrm{T}} = [u_1 \quad u_2 \quad \cdots \quad u_k] \begin{bmatrix} \sigma_1 & 0 & 0 & 0 \\ 0 & \sigma_2 & 0 & 0 \\ 0 & 0 & \ddots & 0 \\ 0 & 0 & 0 & \sigma_k \end{bmatrix} \begin{bmatrix} v_1^{\mathrm{T}} \\ v_2^{\mathrm{T}} \\ \vdots \\ v_k^{\mathrm{T}} \end{bmatrix}$$

矩阵奇异值分解算法

- （3）话题向量空间

- 在单词—文本矩阵X的截断奇异值分解式中，矩阵U_k的每一个列向量 $u_1, u_2, \cdots, u_k$ 表示一个话题，称为话题向量。由这k个话题向量张成一个子空间

$$U_k = \begin{bmatrix} u_1 & u_2 & \cdots & u_k \end{bmatrix}$$

- 称为话题向量空间

矩阵奇异值分解算法

- （4） 文本的话题空间表示
- 有了话题向量空间，接着考虑文本在话题空间的表示

$$
\begin{aligned}
X &= \begin{bmatrix} x_1 & x_2 & \cdots & x_n \end{bmatrix} \approx U_k \Sigma_k V_k^{\mathrm{T}} \\
&= \begin{bmatrix} u_1 & u_2 & \cdots & u_k \end{bmatrix}
\begin{bmatrix} \sigma_1 & & & \\ & \sigma_2 & & 0 \\ & & \ddots & \\ & 0 & & \sigma_k \end{bmatrix}
\begin{bmatrix} v_{11} & v_{21} & \cdots & v_{n1} \\ v_{12} & v_{22} & \cdots & v_{n2} \\ \vdots & \vdots & & \vdots \\ v_{1k} & v_{2k} & \cdots & v_{nk} \end{bmatrix} \\
&= \begin{bmatrix} u_1 & u_2 & \cdots & u_k \end{bmatrix}
\begin{bmatrix} \sigma_1 v_{11} & \sigma_1 v_{21} & \cdots & \sigma_1 v_{n1} \\ \sigma_2 v_{12} & \sigma_2 v_{22} & \cdots & \sigma_2 v_{n2} \\ \vdots & \vdots & & \vdots \\ \sigma_k v_{1k} & \sigma_k v_{2k} & \cdots & \sigma_k v_{nk} \end{bmatrix}
\end{aligned} \tag{17.14}
$$

其中 $u_l = \begin{bmatrix} u_{1l} \\ u_{2l} \\ \vdots \\ u_{ml} \end{bmatrix}, \quad l = 1, 2, \cdots, k$

矩阵奇异值分解算法

- 由式(17.14)知，　矩阵X的第j列向量 x_j 满足

- $(\Sigma_k V_k^{\mathrm{T}})_j$ 是矩阵$(\Sigma_k V_k^{\mathrm{T}})$ 第j列向量
- 式(17.15)是文本 d_j 的近似表达式，
  由 k个话题向量 u_l 的线性组合构成

$$x_j \approx U_k(\Sigma_k V_k^{\mathrm{T}})_j$$

$$= \begin{bmatrix} u_1 & u_2 & \cdots & u_k \end{bmatrix} \begin{bmatrix} \sigma_1 v_{j1} \\ \sigma_2 v_{j2} \\ \vdots \\ \sigma_k v_{jk} \end{bmatrix}$$

$$= \sum_{l=1}^{k} \sigma_l v_{jl} u_l, \quad j = 1, 2, \cdots, n \qquad (17.15)$$

矩阵奇异值分解算法

- 矩阵 $(\Sigma_k V_k^{\mathrm{T}})$ 的每一个列向量

$$\begin{bmatrix} \sigma_1 v_{11} \\ \sigma_2 v_{12} \\ \vdots \\ \sigma_k v_{1k} \end{bmatrix}, \begin{bmatrix} \sigma_1 v_{21} \\ \sigma_2 v_{22} \\ \vdots \\ \sigma_k v_{2k} \end{bmatrix}, \cdots, \begin{bmatrix} \sigma_1 v_{n1} \\ \sigma_2 v_{n2} \\ \vdots \\ \sigma_k v_{nk} \end{bmatrix}$$

- 是一个文本在话题向量空间的表示
- 综上，可以通过对单词一文本矩阵的奇异值分解进行潜在语义分析 $X \approx U_k \Sigma_k V_k^{\mathrm{T}} = U_k(\Sigma_k V_k^{\mathrm{T}})$ 得到话题空间U_k，以及文本在话题空间的表示 $(\Sigma_k V_k^{\mathrm{T}})$

例

- 假设有9个文本，11个单词，单词一文本矩 阵x为11 x 9矩阵，矩阵的元素是单词在文本中出现的频数，表示如下：

Index Words	Titles								
	T1	T2	T3	T4	T5	T6	T7	T8	T9
book			1	1					
dads						1			1
dummies		1						1	
estate							1		1
guide	1					1			
investing	1	1	1	1	1	1	1	1	1
market	1		1						
real							1		1
rich						2			1
stock	1		1					1	
value				1	1				

- 进行潜在语义分析。

例

- 实施对矩阵的截断奇异值分解，假设话题的个数是3，截断奇异值分解结果为

Book	0.15	−0.27	0.04
Dads	0.24	0.38	−0.09
Dummies	0.13	−0.17	0.07
Estate	0.18	0.19	0.45
Guide	0.22	0.09	−0.46
Investing	0.74	−0.21	0.21
Market	0.18	−0.30	−0.28
Real	0.18	0.19	0.45
Rich	0.36	0.59	−0.34
Stock	0.25	−0.42	−0.28
Value	0.12	−0.14	0.23

$*$

3.91	0	0
0	2.61	0
0	0	2.00

$*$

T1	T2	T3	T4	T5	T6	T7	T8	T9
0.35	0.22	0.34	0.26	0.22	0.49	0.28	0.29	0.44
−0.32	−0.15	−0.46	−0.24	−0.14	0.55	0.07	−0.31	0.44
−0.41	0.14	−0.16	0.25	0.22	−0.51	0.55	0.00	0.34

例

- 左矩阵U_3个列向量（左奇异向量）。第1列向量 u_1 的值均为正，第2列向量 u_2 和第3列向量 u_3 的值有正有负。

- 中间的对角矩阵 Σ_3 的元素是3个由 大到小的奇异值（正值）。

- 右矩阵是 V_3^{T} ，其转置矩阵V_3也有3个列向量（右奇异向 量）。第1列向量 v_1 的值也都为正，第2列向量 v_2 和第3列向量 v_3 的值有正有负。

例

- 现在，将 Σ_3 与 V_3^{T} 相乘，整体变成两个矩阵乘积的形式

$$X \approx U_3(\Sigma_3 V_3^{\mathrm{T}})$$

$$= \begin{bmatrix} 0.15 & -0.27 & 0.04 \\ 0.24 & 0.38 & -0.09 \\ 0.13 & -0.17 & 0.07 \\ 0.18 & 0.19 & 0.45 \\ 0.22 & 0.09 & -0.46 \\ 0.74 & -0.21 & 0.21 \\ 0.18 & -0.30 & -0.28 \\ 0.18 & 0.19 & 0.45 \\ 0.36 & 0.59 & -0.34 \\ 0.25 & -0.42 & -0.28 \\ 0.12 & -0.14 & 0.23 \end{bmatrix} \begin{bmatrix} 1.37 & 0.86 & 1.33 & 1.02 & 0.86 & 1.92 & 1.09 & 1.13 & 1.72 \\ -0.84 & -0.39 & -1.20 & -0.63 & -0.37 & 1.44 & 0.18 & -0.81 & 1.15 \\ -0.82 & 0.28 & -0.32 & 0.50 & 0.44 & -1.02 & 1.10 & 0.00 & 0.68 \end{bmatrix}$$

例

- 矩阵U_3有3个列向量，表示3个话题，矩阵U_3表示话题向量空间。

- 矩阵$(\Sigma_3 V_3^{\mathrm{T}})$有9个列向量，表示9个文本，矩阵$(\Sigma_3 V_3^{\mathrm{T}})$是文本集合在话题向量空间的表示。

非负矩阵分解算法

- 非负矩阵分解也可以用于话题分析。
- 对单词—文本矩阵进行非负矩阵分解，将其左矩阵作为话题向量空间，将其右矩阵作为文本在话题向量空间的表示。
- 注意通常单词-文本矩阵是非负的。

非负矩阵分解

- 给定一个非负矩阵X≥0，找到两个非负矩阵W≥0和H≥ 0，使得

$$X \approx WH$$

- 即将非负矩阵X分解为两个非负矩阵W和H的乘积的形式，称为非负矩阵分解。

- 因为WH与X完全相等很难实现，所以只要求WH与X近似相等。

非负矩阵分解

- 假设非负矩阵X是 m x n 矩阵，非负矩阵W和H分别为 m x k 矩阵和 k x n 矩阵。

- 假设k<min(m, n)，即W和H小于原矩阵X，所以非负矩阵分解是对原数据的压缩。

非负矩阵分解

- 由 $X \approx WH$ 知，矩阵X的第j列向量 x_j 满足

$$x_j \approx Wh_j$$

$$= \begin{bmatrix} w_1 & w_2 & \cdots & w_k \end{bmatrix} \begin{bmatrix} h_{1j} \\ h_{2j} \\ \vdots \\ h_{kj} \end{bmatrix} \qquad = \sum_{l=1}^{k} h_{lj} w_l, \quad j = 1, 2, \cdots, n$$

- 矩阵X的第j列 x_j 可以由矩阵W的k个列 w_l 的线性组合逼近，线性组合的系数是矩阵H的第j列h_j的元素。
- 非负矩阵分解旨在用较少的基向量、系数向量来表示较大的数据矩阵。

潜在语义分析模型

- 给定一个 m x n 非负的单词-文本矩阵X≥0

- 假设文本集合共包含k个话题， 对X进行非负矩阵分解。即求非负的 m x k 矩阵W≥0和 k x n 矩阵H≥0，使得

$$X \approx WH$$

- 令 $W = [w_1 \quad w_2 \quad \cdots \quad w_k]$ 为话题向量空间，$w_1, w_2, \cdots, w_k$ 表示文本集合的k个 话题，令 $H = [h_1 \quad h_2 \quad \cdots \quad h_n]$ 为文本在话题向量空间的表示，$h_1, h_2, \cdots, h_n$ 表示 文本集合的n个文本

非负矩阵分解的形式化

- 非负矩阵分解可以形式化为最优化问题求解。首先定义损失函数或代价函数。

- 第一种损失函数是平方损失。设两个非负矩阵 $A = [a_{ij}]_{m \times n}$ ，和 $B = [b_{ij}]_{m \times n}$ ，平方损失函数定义为

$$\|A - B\|^2 = \sum_{i,j} (a_{ij} - b_{ij})^2$$

- 其下界是0，当且仅当A=B时达到下界。

非负矩阵分解的形式化

- 另一种损失函数是散度（divergence)。设两个非负矩阵 $A = [a_{ij}]_{m\times n}$ 和 $B = [b_{ij}]_{m\times n}$ 散度损失函数定义为

$$D(A\|B) = \sum_{i,j}\left(a_{ij}\log\frac{a_{ij}}{b_{ij}} - a_{ij} + b_{ij}\right)$$

- 其下界也是0，当且仅当A= B时达到下界。A和B不对称。

- 当 $\sum_{i,j} a_{ij} = \sum_{i,j} b_{ij} = 1$ 时
- 散度损失函数退化为Kuliback-Leiber散度或相对熵，这时A和B是概率分布。

非负矩阵分解的形式化

- 目标函数: $\|X - WH\|^2$ 关于W和H的最小化，满足约束条件W, H≥0, 即

$$\min_{W,H} \quad \|X - WH\|^2$$
$$\text{s.t.} \quad W, H \geqslant 0$$

- 或者，目标函数 $D(X\|WH)$ 关于W和H的最小化，满足约束条件W, H≥0，即

$$\min_{W,H} \quad D(X\|WH)$$
$$\text{s.t.} \quad W, H \geqslant 0$$

算法

定理 17.1 平方损失 $\|X - WH\|^2$ 对下列乘法更新规则

$$H_{lj} \leftarrow H_{lj} \frac{(W^{\mathrm{T}}X)_{lj}}{(W^{\mathrm{T}}WH)_{lj}} \tag{17.24}$$

$$W_{il} \leftarrow W_{il} \frac{(XH^{\mathrm{T}})_{il}}{(WHH^{\mathrm{T}})_{il}} \tag{17.25}$$

是非增的。当且仅当 W 和 H 是平方损失函数的稳定点时函数的更新不变。

算法

定理 17.2 散度损失 $D(X - WH)$ 对下列乘法更新规则

$$H_{lj} \leftarrow H_{lj} \frac{\sum_i [W_{il} X_{ij} / (WH)_{ij}]}{\sum_i W_{il}} \tag{17.26}$$

$$W_{il} \leftarrow W_{il} \frac{\sum_j [H_{lj} X_{ij} / (WH)_{ij}]}{\sum_j H_{lj}} \tag{17.27}$$

是非增的。当且仅当 W 和 H 是散度损失函数的稳定点时函数的更新不变。

算法

- 最优化目标函数是 $\|X - WH\|^2$，为了方便将目标函数乘以1/2，其最优解与原问 题相同，记作

$$J(W,H) = \frac{1}{2}\|X - WH\|^2 = \frac{1}{2}\sum[X_{ij} - (WH)_{ij}]^2$$

- 应用梯度下降法求解。首先求目标函数的梯度

$$\frac{\partial J(W,H)}{\partial W_{il}} = -\sum_j [X_{ij} - (WH)_{ij}]H_{lj}$$

$$= -[(XH^{\mathrm{T}})_{il} - (WHH^{\mathrm{T}})_{il}]$$

- 同样可得

$$\frac{\partial J(W,H)}{\partial H_{lj}} = -[(W^{\mathrm{T}}X)_{lj} - (W^{\mathrm{T}}WH)_{lj}]$$

算法

- 然后求得梯度下降法的更新规则

$$W_{il} = W_{il} + \lambda_{il}[(XH^{\mathrm{T}})_{il} - (WHH^{\mathrm{T}})_{il}]$$

$$H_{lj} = H_{lj} + \mu_{lj}[(W^{\mathrm{T}}X)_{lj} - (W^{\mathrm{T}}WH)_{lj}]$$

- 式中 λ_{il}, μ_{lj} 是步长。选取

$$\lambda_{il} = \frac{W_{il}}{(WHH^{\mathrm{T}})_{il}}, \quad \mu_{lj} = \frac{H_{lj}}{(W^{\mathrm{T}}WH)_{lj}}$$

- 即得乘法更新规则

$$W_{il} = W_{il}\frac{(XH^{\mathrm{T}})_{il}}{(WHH^{\mathrm{T}})_{il}}, \quad i = 1, 2, \cdots, m; \quad l = 1, 2, \cdots, k$$

$$H_{lj} = H_{lj}\frac{(W^{\mathrm{T}}X)_{lj}}{(W^{\mathrm{T}}WH)_{lj}}, \quad l = 1, 2, \cdots, k; \quad j = 1, 2, \cdots, n$$

非负矩阵分解的迭代算法

输入：单词-文本矩阵 $X \geqslant 0$，文本集合的话题个数 k，最大迭代次数 t；

输出：话题矩阵 W，文本表示矩阵 H。

（1）初始化

$W \geqslant 0$，并对 W 的每一列数据归一化；

$H \geqslant 0$；

（2）迭代

对迭代次数由 1 到 t 执行下列步骤：

（a）更新 W 的元素，对 l 从 1 到 k，i 从 1 到 m 按式 (17.33) 更新 W_{il}；

（b）更新 H 的元素，对 l 从 1 到 k，j 从 1 到 n 按式 (17.34) 更新 H_{lj}。