



LDA (Latent Dirichlet Allocation)

模型的原理及其应用

哈尔滨工业大学
智能技术与自然语言处理实验室

报告人：轩文烽

2010-12-15

提纲

- 背景
- 准备知识
- LDA模型的原理
- 一个简单应用
- Further Reading
- 一些有用的资源
- 主要参考资料

背景

对于给定的一个文档集合 我们相知道:

- TREC2007
- 它
- 每
- 不

```
<top>
  <num> Number: 994 </num>

  <title> formula f1 </title>

  <desc> Description:
  Blogs with interest in the formula
  one (f1) motor racing, perhaps with
  driver news, team news, or event
  news.
  </desc>

  <narr> Narrative:
  Relevant blogs will contain news
  and analysis from the Formula f1
  motor racing circuit. Blogs with
  documents not in English are not
  relevant.
  </narr>
</top>
```

1)

* 直观: 文档集合的内容体现了多个主题

准备知识

- 概率分布
- Bayesian Network
- Expectation-Maximization (EM) 算法
- Variational Inference

概率分布—多项分布

- Question: 投掷一枚**硬币**，可能的结果有两种：正面向上(1)和反面向上(0)。现用随机变量 Y 来表示这些结果，并设 $P(Y=1)=p$ 。那么投掷 N 次硬币，其中 k 次为正面向上的概率是多少？

$$\binom{N}{k} p^k (1-p)^{N-k} = \frac{N!}{k!(N-k)!} p^k q^{N-k}$$

$$q = 1 - p \Rightarrow p + q = 1$$

$$X = (k, N - k)$$

- 推广：现在投掷N次骰子，已知在每次投掷中*i*向上的概率为 p_i ， $i = 1, 2, \dots, 6$. 且 $\sum_{i=1}^6 p_i = 1$. 那么在N次中*i*出现 n_i 次， $i = 1, 2, \dots, 6$ 的概率是多少？

$$\frac{N!}{\prod_{i=1}^6 n_i!} \left(\prod_{i=1}^6 p_i^{n_i} \right)$$

$$\sum_{i=1}^6 p_i = 1$$

$$\sum_{i=1}^6 n_i = N$$

$$X = (n_1, n_2, \dots, n_6)$$

掷硬币

$$\frac{N!}{k!(N-k)!} p^k q^{N-k} \quad p+q=1 \quad X=(k, N-k)$$

掷骰子

$$\frac{N!}{\prod_{i=1}^6 n_i} \left(\prod_{i=1}^6 p_i^{n_i} \right) \quad \sum_{i=1}^6 p_i = 1 \quad \sum_{i=1}^6 n_i = N \quad X=(n_1, n_2, \dots, n_6)$$

多项分布
(Multinomial
Distribution)

$$\frac{N!}{\prod_{i=1}^K n_i} \left(\prod_{i=1}^K p_i^{n_i} \right) \quad \sum_{i=1}^K p_i = 1 \quad \sum_{i=1}^K n_i = N \quad X=(n_1, n_2, \dots, n_K)$$

$$(p_1 + p_2 + \dots + p_K)^N$$

概率分布—Dirichlet分布

- 假设我们在和一个人玩掷骰子游戏。正常情况下我们都会认为骰子的每个面出现的概率是相等的，为 $1/6$ ；但是现在我们看到掷骰子的人连续掷出6，不免心生猜测：

50%的可能：6出现的概率为 $2/7$ ，其他各面为 $1/7$ ；

25%的可能：6出现的概率为 $3/8$ ，其他各面为 $1/8$ ；

25%的可能：各面的概率为 $1/6$

Probability of each face under each hypothesis about how the die is loaded

Belief	Face	1	2	3	4	5	6
.5	Probability	<u>1/7</u>	<u>1/7</u>	<u>1/7</u>	<u>1/7</u>	<u>1/7</u>	<u>2/7</u>
.25	Probability	1/8	1/8	1/8	1/8	1/8	3/8
.25	Probability	1/6	1/6	1/6	1/6	1/6	1/6

如果记我们所猜测的每个面出现的概率为 X ，那么表示 X 的最恰当分布就是狄立克雷分布(Dirichlet Distribution)。

$$p(X | \alpha) = \frac{\Gamma\left(\sum_{i=1}^K \alpha_i\right)}{\prod_{i=1}^K \Gamma(\alpha_i)} \left(\prod_{i=1}^K x_i^{\alpha_i - 1} \right)$$

$$X = (x_1, x_2, \dots, x_K) \quad \sum_{i=1}^K x_i = 1 \quad \alpha = (\alpha_1, \alpha_2, \dots, \alpha_K)$$

概率分布—The Exponential Family

- 在指数分布族中，随机变量(或向量)的概率分布(或密度)函数具有如下形式：

$$p(x|\eta) = h(x)g(\eta)\exp\{\eta^T u(x)\}$$

其中， η 为参数， $g(\eta)$ 为归一化因子

概率分布—共轭先验 (Conjugate Prior)


- 对于概率分布(或密度)函数 $p(x|\eta)$, 若 $p(\eta)$ 满足如下条件, 则称 $p(\eta)$ 为 $p(x|\eta)$ 的共轭先验:

(1) 后验分布 $p(\eta|x)$ 与 $p(\eta)$ 有相同的函数形式。

- 指数分布族中的每一个成员均具有如下形式的共轭先验:

$$p(\eta|\chi, \nu) = f(\chi, \nu) g(\eta)^\nu \exp\{\nu \eta^T \chi\}$$

狄立克雷分布是多项分布的共轭先验


$$p(\eta | x) = \frac{p(\eta) p(x | \eta)}{p(x)}$$

概率分布—可交换性及 de Finetti 定理

- 可交换性：随机变量 z_1, z_2, \dots, z_n 称为是可交换的，如果满足如下条件：

$$p(z_1, z_2, \dots, z_n) = p(z_{\pi(1)}, z_{\pi(2)}, \dots, z_{\pi(n)})$$

- 无限可交换性：随机变量的无限序列 z_1, z_2, \dots 称为是无限可交换的，如果其中任何一个有限子序列都是可交换的

de Finetti 定理

More precisely, suppose X_1, X_2, X_3, \dots is an infinite exchangeable sequence of Bernoulli-distributed random variables. Then there is some probability distribution m on the interval $[0, 1]$ and some random variable Y such that

- The probability distribution of Y is m , and
- The **conditional probability distribution** of the whole sequence X_1, X_2, X_3, \dots given the value of Y is described by saying that
 - X_1, X_2, X_3, \dots are **conditionally independent** given Y , and
 - For any $i \in \{1, 2, 3, \dots\}$, the conditional probability that $X_i = 1$, given the value of Y , is Y .

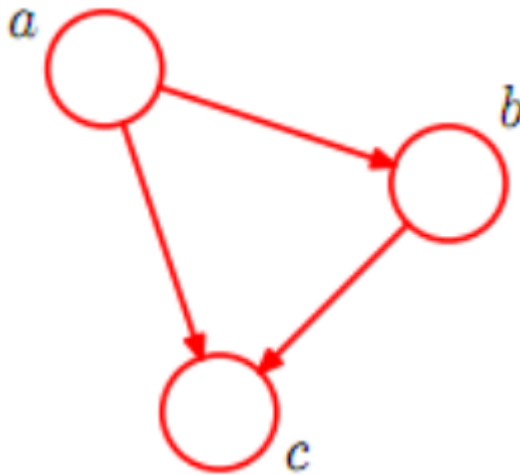
$$p(X_1, X_2, \dots) = \int_y p(y) \left(\prod_i p(X_i | y) \right) dy$$

$$p(X_i = 1 | y) = y$$

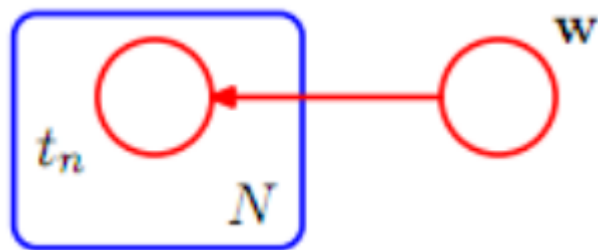
多项分布!

Bayesian Network

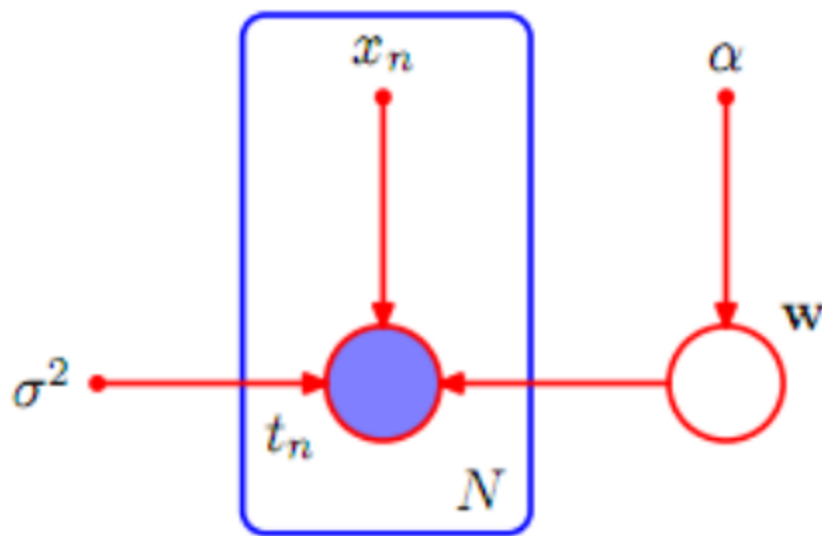
- 对于联合概率 $P(a,b,c) = P(a)P(b|a)P(c|a,b)$, 如何用图形进行表示呢?



- 复杂一点：对于联合概率 $P(t, w) = P(w) \prod_{n=1}^N P(t_n | w)$ 怎么表示呢？



- 再复杂一点：对于一些概率模型，有些变量是我们观测到的，有些则是我们需要估计的，这两种节点有必要在图中区分开来



EM算法

- EM (Expectation-Maximization) 算法是一种用于求解含有隐含变量的模型的极大似然解的方法

单个高斯分布

- 假设我们有一个观测数据集 $\{x_1, x_2, \dots, x_N\}$, 我们想用一个小高斯分布来对该数据集进行建模

$$(\mu^*, \Sigma^*) = \arg \max \prod_{n=1}^N p(x_n | \mu, \Sigma)$$

$$\mu^* = \frac{1}{N} \sum_{n=1}^N x_n \quad \Sigma^* = \frac{1}{N} \sum_{n=1}^N (x_n - \mu^*)(x_n - \mu^*)^T$$

多个高斯分布——高斯混合模型

- 现在我们改用一个高斯混合模型来对上面的数据集进行建模

$$p(x) = \sum_{k=1}^K \pi_k p(x | \mu_k, \Sigma_k)$$

$$\prod_{n=1}^N \left(\sum_{k=1}^K \pi_k p(x_n | \mu_k, \Sigma_k) \right)$$

$$\mu_k^* = \frac{1}{N_k} \sum_{n=1}^N \gamma(z_{nk}) x_n \quad \Sigma_k^* = \frac{1}{N_k} \sum_{n=1}^N \gamma(z_{nk}) (x_n - \mu_k^*) (x_n - \mu_k^*)^T \quad \pi_k^* = \frac{N_k}{N}$$

$$\gamma(z_{nk}) = \frac{\pi_k p(x_n | \mu_k, \Sigma_k)}{\sum_{j=1}^K \pi_j p(x_n | \mu_j, \Sigma_j)} \quad N_k = \sum_{n=1}^N \gamma(z_{nk})$$

依赖关系

$$\mu_k^* = \frac{1}{N_k} \sum_{n=1}^N \gamma(z_{nk}) x_n \quad \Sigma_k^* = \frac{1}{N_k} \sum_{n=1}^N \gamma(z_{nk}) (x_n - \mu_k^*) (x_n - \mu_k^*)^T \quad \pi_k^* = \frac{N_k}{N}$$

$$\mu_k \leftarrow \gamma(z_{nk}) \leftarrow \pi_k, \Sigma_k, \mu_k$$

$$\Sigma_k \leftarrow \gamma(z_{nk}) \leftarrow \pi_k, \mu_k, \Sigma_k$$

$$\pi_k \leftarrow N_k \leftarrow \gamma(z_{nk}) \leftarrow \mu_k, \Sigma_k, \pi_k$$

相互依赖

迭代求解!

混合高斯中的隐含变量

$$p(x) = \sum_{k=1}^K \pi_k p(x | \mu_k, \Sigma_k)$$

$$p(z_k = 1) = \pi_k \quad \Rightarrow \quad p(z) = \prod_{k=1}^K \pi_k^{z_k}$$

$$p(x | z_k = 1) = p(x | \mu_k, \Sigma_k) \quad \Rightarrow \quad p(x | z) = \prod_{k=1}^K p(x | \mu_k, \Sigma_k)^{z_k}$$

$$p(x) = \sum_z p(z) p(x | z) = \sum_z p(x, z)$$

z 就是隐含变量

EM的一般过程

给定联合分布 $p(X, Z | \theta)$ ，其中 X 为观测到的变量， Z 为隐含变量， θ 为参数，以下过程用来求解似然函数 $p(X | \theta)$ 的极大值：

- ✓ 设定参数的初始值 θ^{old}
- ✓ E step: 计算 $p(Z | X, \theta^{old})$
- ✓ M step: 计算 $\theta^{new} = \arg \max Q(\theta, \theta^{old})$

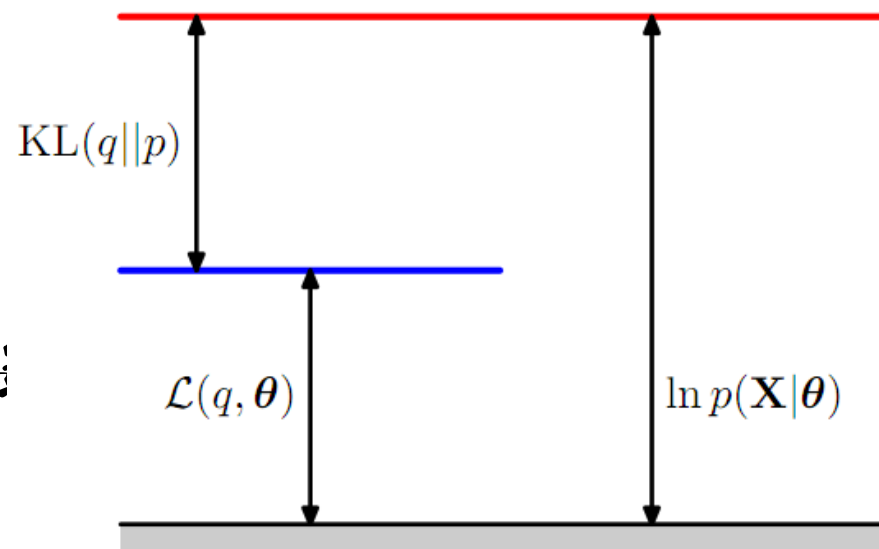
$$Q(\theta, \theta^{old}) = \sum_Z p(Z | X, \theta^{old}) \ln p(X, Z | \theta)$$

- ✓ 判断似然函数或者参数值是否收敛，不收敛时进行如下更新：

$$\theta^{old} \leftarrow \theta^{new}$$

为什么是这样

- 对于任意的分布 q 成立：



$$\ln p(\mathbf{X} | \theta) = L(q, \theta) + KL(q || p)$$

$$L(q, \theta) = \sum_z q(Z) \ln \left\{ \frac{p(\mathbf{X}, Z | \theta)}{q(Z)} \right\}$$

$$KL(q || p) = - \sum_z q(Z) \ln \left\{ \frac{p(Z | \mathbf{X}, \theta)}{q(Z)} \right\}$$

- 令 $q(Z) = p(Z | X, \theta^{old})$, 我们可以得到

$$\begin{aligned} L(q, \theta) &= \sum_Z p(Z | X, \theta^{old}) \ln p(X, Z | \theta) \\ &\quad - \sum_Z p(Z | X, \theta^{old}) \ln p(Z | X, \theta^{old}) \\ &= Q(\theta, \theta^{old}) + const \end{aligned}$$

在迭代的过程中, 似然函数的值是单调增加的

Variational Inference

- 变分推理是一种用来近似计算后验概率的方法。
- 对于EM算法中的E step, 我们是通过令 $q(Z) = p(Z|X, \theta^{old})$ 来得到 $L(q, \theta)$ 的极大值的。如果这个后验概率的计算很困难, 那么我们该怎么办呢?

限制 $q(Z)$ 的可选范围来近似求解

$$q(Z) = \prod_{i=1}^M q_i(Z_i)$$

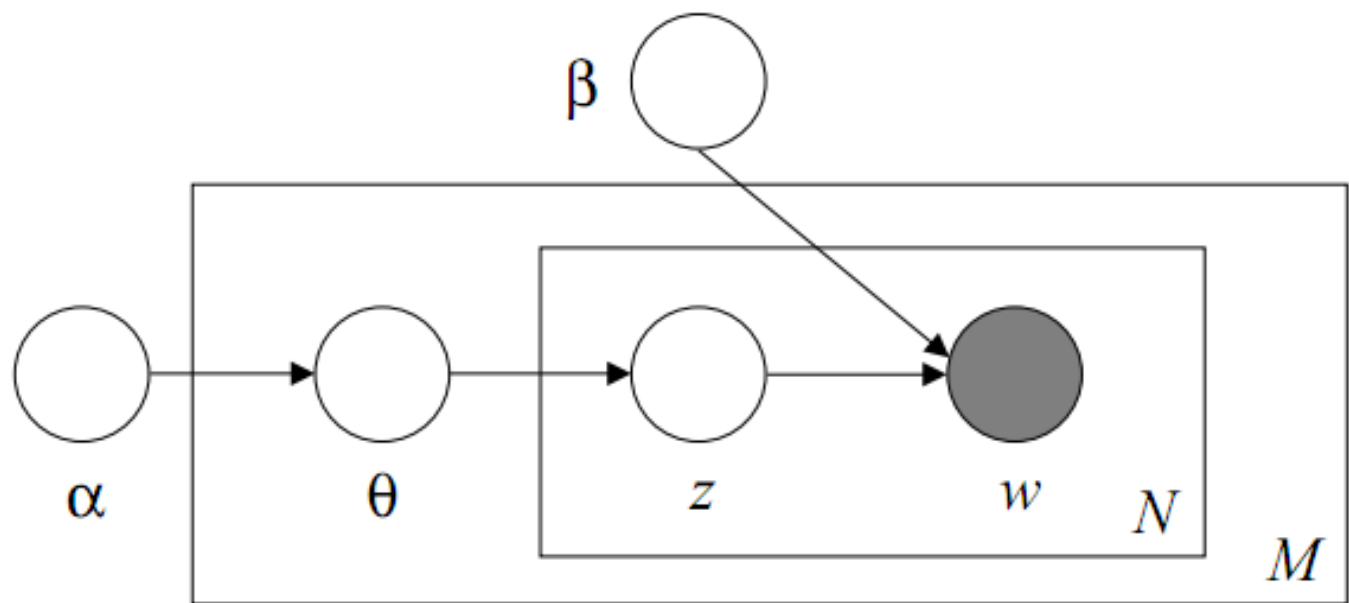
LDA模型的原理

- LDA (Latent Dirichlet Allocation)模型希望通过将文档表示为一个主题向量来达到特征降维的目的
- 假设文档、文档中词的顺序是无关紧要的
- 符号说明：
 - ✓ 词表大小 V ，每个词用一个 V 维向量进行表示 $(0, 1, 0, \dots, 0)$
 - ✓ 一个由 N 个词构成的文档记为 $w = (w_1, w_2, \dots, w_N)$
 - ✓ 一个由 M 篇文档构成的语料记为 $D = \{w_1, w_2, \dots, w_M\}$

一篇文档的生成过程：

- 第一步：选择文档长度 N , $N \sim \text{Poisson}(\xi)$
- 第二步：选择 θ , $\theta \sim \text{Dirichlet}(\alpha)$, 这里 θ 是矢量，表示每个主题发生的概率， α 是 *Dirichlet* 分布的参数
- 第三步：对 N 个单词中每一个：
 - ✓ 选择主题 z_n , $z_n \sim \text{Multinomial}(\theta)$
 - ✓ 选择 w_n , 根据多项分布 $P(w_n | z_n, \beta)$

图形化表示



模型的执行过程

主要分为两步：

- 训练：进行参数估计

$$(\alpha^*, \beta^*) = \arg \max \prod_{d=1}^M p(w_d | \alpha, \beta)$$

- 测试：计算后验分布

$$p(\theta, z | w, \alpha, \beta)$$

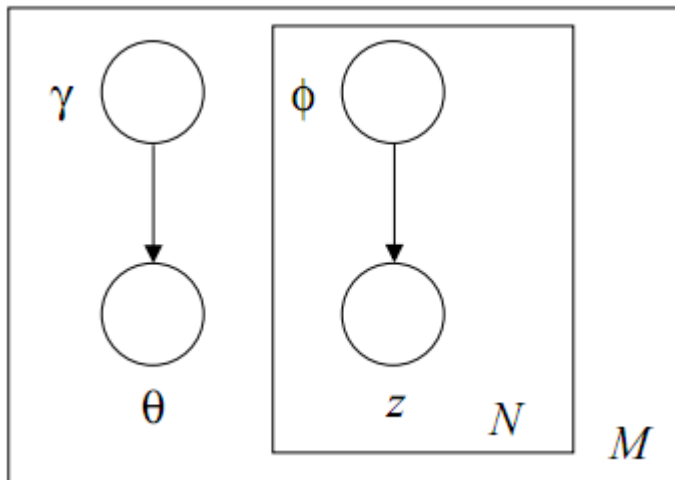
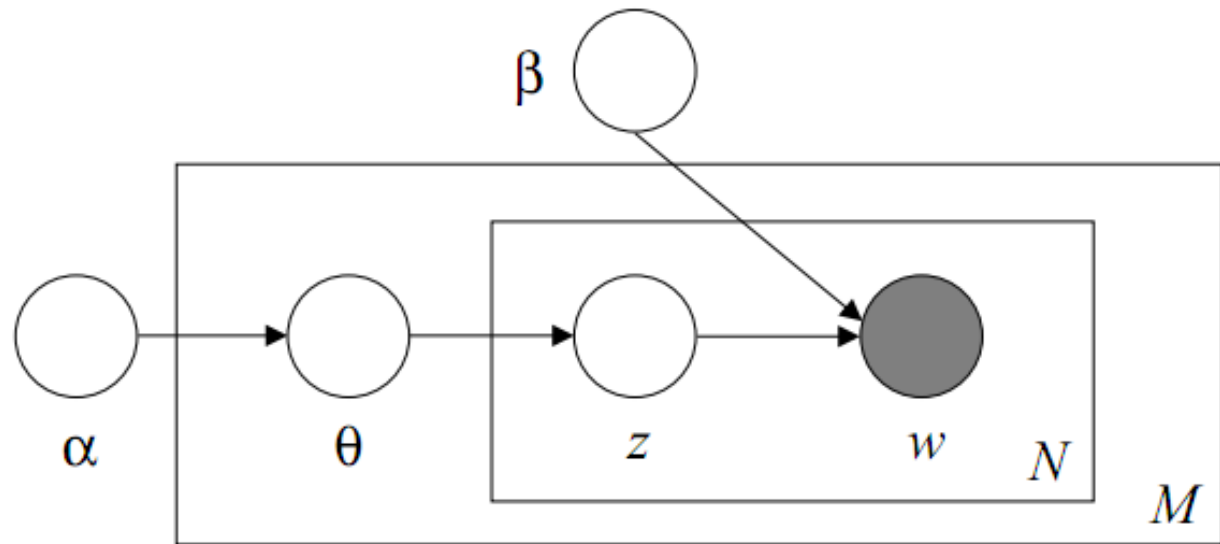
后验分布的计算

$$p(\theta, z | w, \alpha, \beta) = \frac{p(\theta, z, w | \alpha, \beta)}{p(w | \alpha, \beta)}$$

$$\begin{aligned} p(w | \alpha, \beta) &= \int p(\theta | \alpha) \left(\prod_{n=1}^N \sum_{z_n} p(z_n | \theta) p(w_n | z_n, \beta) \right) d\theta \\ &= \frac{\Gamma(\sum_i \alpha_i)}{\prod_i \Gamma(\alpha_i)} \int \left(\prod_{i=1}^k \theta_i^{\alpha_i - 1} \right) \left(\prod_{n=1}^N \sum_{i=1}^k \prod_{j=1}^V (\theta_i \beta_{ij})^{w_n^j} \right) d\theta \end{aligned}$$

直接计算很难，怎么办？

近似



$$q(\theta, z | \gamma, \phi) = q(\theta | \gamma) \prod_{n=1}^N q(z_n | \phi_n)$$

$$(\gamma^*, \phi^*) = \arg \min KL(q(\theta, z | \gamma, \phi) \| p(\theta, z | w, \alpha, \beta))$$

算法描述

initialize $\phi_{ni}^0 := 1/k$ for all i and n

initialize $\gamma_i := \alpha_i + N/k$ for all i

repeat

for $n = 1$ **to** N

for $i = 1$ **to** k

$\phi_{ni}^{t+1} := \beta_{iw_n} \exp(\Psi(\gamma_i^t))$

 normalize ϕ_n^{t+1} to sum to 1.

$\gamma^{t+1} := \alpha + \sum_{n=1}^N \phi_n^{t+1}$

until convergence

参数估计

$$(\alpha^*, \beta^*) = \arg \max \prod_{d=1}^M p(w_d | \alpha, \beta)$$

- 变分EM算法：

- ✓ 设定 α, β 的初始值

- ✓ E step: 利用 variational inference 计算 γ, ϕ
来近似似然函数

- ✓ M step: 根据 α, β 极大化 E step 中的结果

- ✓ 结束准则判断

LDA模型是怎么得出来的？

假设文档、文档中的词的顺序无关紧要

可交换性 de Finetti 定理

$$p(w|\alpha, \beta) = \int p(\theta|\alpha) \left(\prod_{n=1}^N \sum_{z_n} p(z_n|\theta) p(w_n|z_n, \beta) \right) d\theta$$

$p(z_n|\theta)$ 是多项分布 ← de Finetti 定理

$p(\theta|\alpha)$ 是 Dirichlet 分布 ← 共轭先验

$p(w_n|z_n, \beta)$ 是多项分布 ← 指数分布族

参数估计，后验分布计算

一个简单应用

- 将LDA模型应用于1篇文档，进行关键词抽取
- ✓ 依据：LDA模型能够得到每个主题生成每个词的概率，那么我们就可以把每个主题中的TOPK个词取出来作为该文档的关键词(移除重复出现的词)

步骤

- 第一步：准备数据

- ✓ word_frequent.txt

- 格式：[word num] [wordid:occ_num] [...]

- ✓ words.txt

- 格式：每行一个词

- 第二步：模型计算

- ✓ lda est [alpha] [k] [settings] [data]

- [random/seeded/*] [directory]

- ✓ 输出：final.other, final.beta, final.gamma, word-assignments.dat

- 第三步：获取关键词

实验结果

Puppet: 网络数据中心自动化配置管理				
tfidf	ictclas	lda_2topic	lda_3topic	lda_4topic
Puppet	系统	Puppet	Puppet	Puppet
审计	配置	管理	系统	系统
设施	管理	配置	管理	管理
社区	一个	系统	配置	配置
成千上万	基础	一个	一个	一个
陈述	社区	社区	社区	基础
活跃	提供	基础	中心	数据
数百	数据	使用	基础	中心
自动化	模块	数据	模块	社区
Puppet:	用户	设施	用户	用户
PuppetSecure	中心	模块	帮助	提供
Puppeties	模型	支持	利用	帮助
Puppetmaster	设施	用户	模型	支持
SuSE	利用	中心	优势	模型
机架	帮助	提供	数据	集中
Forge	使用	状态	状态	模块
Fedora	需要	利用	管理员	控制
introduction	支持	组织	使用	需要
Debian	状态	需要	集中	状态
www.puppetlabs.com	平台	图形	提供	执行
CentOS	模式	开源	操作	平台
2010.10.11	控制	模型	资源	活跃
中心	开源	帮助	支持	设施
不须	集中	知道	是否	成千上万
纠偏	操作	资源	开源	是否
puppet	灵活	优势	Unix	知道
红帽	操作系统	自动化	设施	操作系统
配置	测试	活跃	希望	陈述
Unix	当前	容易	自动化	使用
审批	问题	数百	需要	问题

Mac OS X: 修改SMART Utility期限限制

tfidf	ictclas	lda_2topic	lda_3topic	lda_4topic
S. M. A. R. T	软件	软件	使用	里面
SMART	使用	使用	软件	文件
Libraray	里面	里面	信息	软件
Utility	一个	技术	里面	使用
措施	文件	没有	限制	信息
保密	信息	一个	一个	一个
貌似	当前	问题	文件	Utility
硬盘	技术	硬盘	技术	技术
加密	限制	信息	加密	没有
com.volitans-software.smartutility.plist	问题	文件	需要	硬盘
com.apple.services	措施	看到	硬盘	限制
Volitans-Software	硬盘	S. M. A. R. T	没有	问题
Verified	没有	需要	措施	措施
Registration.plist	需要	加密	问题	S. M. A. R. T
手气	注册	措施	保密	需要
安全系数	加密	限制	开发	加密
苹果	看到	注册	Utility	看到
限制	提供	程序	提供	简单
镇密	提高	Utility	注册	貌似
探讨	探讨	保密	苹果	注册
试用期	是否	Preferences	方面	当前
Preferences	上面	安全	容易	Library
注册	容易	简单	看到	可能
出品	开发	考虑	上面	保密
软件	程序	来说	提高	苹果
里面	后来	开发	S. M. A. R. T	方面
Disk	今天	貌似	可能	Preferences
欠缺	保密	苹果	当前	上面
试用	方面	可能	SMART	开发
Support	可能	提高	Library	SMART

Further Reading

- Correlated Topic Models. Neural Information Processing Systems, 2006
- Dynamic topic models. In Proceedings of the 23rd International Conference on Machine Learning, 2006
- Online Learning for Latent Dirichlet Allocation. Neural Information Processing Systems, 2010
- Markov Chain Monte Carlo and Gibbs Sampling. 2004

一些有用的资源

Topic Model领域的一些大牛:

- DM Blei (LDA的提出者):

<http://www.cs.princeton.edu/~blei/>

- Thomas Hofmann (pLSA的提出者):

<http://www.cs.brown.edu/~th/>

- Andrew McCallum:

<http://www.cs.umass.edu/~mccallum/>

主要参考资料

- David M. Blei, Andrew Y. Ng, Michael I. Jordan. Latent Dirichlet Allocation. Journal of Machine Learning Research 2003
- David M. Blei, John Lafferty. Modeling Science. 2008
- Zhou Li. Latent dirichlet allocation note
- Steffen Lauritzen. Exchangeability and de Finetti's Theorem. University of Oxford 2007
- Christopher M. Bishop. Pattern Recognition and Machine Learning 第2、8、9、10章



Thank You!