◎数据库、信息处理◎

贝叶斯概率 LSA 模型权重更新算法

曾广平
ZENG Guang-ping

中南民族大学 计算机科学学院,武汉 430074
College of Computer Science,South-Central University for Nationalities,Wuhan 430074,China
E-mail:zenggp09@163.com

ZENG Guang-ping.Bayesian probability weight update algorithm of LSA model.Computer Engineering and Applications, 2009,45(21):88-90.

Abstract: To the weight update of Latent Semantic Analysis(LSA) model,this paper proposes an adaptive weight update algorithm based on Bayesian theory(ALSAB).ALSAB adopts Maximum A Posteriori(MAP) provability estimation and Expectation-Maximization(EM) algorithm to estimate the weight parameters of LSA,and ALSAB employs incremental learning to decrease accumulative effect caused in continuous update with considering that the probability of uncommon words decreases in continuous update.Experimental results show that,compared with the existing algorithms,the proposed ALSAB algorithm greatly improves recall and precision rates of information retrieval systems.

Key words: Latent Semantic Analysis(LSA);Bayesian;weight update

摘 要:针对潜在语义分析(LSA)模型的权重更新问题,提出了一种基于贝叶斯理论的自适应权重更新算法 ALSAB。ALSAB 采用最大后验概率估计与期望值最大(EM)算法对概率 LSA 模型参数进行有效的估计,在充分考虑多次更新中不常用字词概率参数降低问题的前提下,采用增量学习方法降低多次更新产生的累积效应。实验结果表明,与现有的权重更新算法相比,提出的 ALSAB 算法显著地提高了检索的准确率与召回率。

关键词:潜在语义分析;贝叶斯;权重更新
DOI:10.3778/j.issn.1002-8331.2009.21.025 **文章编号:**1002-8331(2009)21-0088-03 **文献标识码:**A **中图分类号:**TP311.12

1 引言

随着信息技术与互联网的飞速发展,如何对海量的信息进行准确有效地检索成为研究的热点[1]。基于潜概念(Latent Concept)索引的潜在语义分析(Latent Semantic Analysis,LSA)[2-3],作为基于概念的检索技术,由于需要人的参与性少、可计算性和可操作性强等优势,已经被证明是对传统的向量空间技术的一种改良,在信息检索、信息过滤、文档分类、自动文摘等方面都取得了很好的应用[4-5]。LSA 基于这样一个假设:词语出现在某一个文档中以及两个词语出现在同一段上下文中不是完全随机的,而是某种潜在语义结构在起作用。如果能把这种潜在语义结构提取出来,建立词与词之间的语义关系,就可以消除词语用法的多样性和词语使用的随意性对检索产生的偏差。LSA 利用奇异值分解(Singular Value Decomposition,SVD)生成的潜概念索引来进行信息检索,而不再是传统的基于检索词匹配。

为了突出各个词语和文档对语义空间不同的贡献程度,LSA 定义了一个权重函数对词频矩阵进行加权转换。由于权重决定了检索的准确性,因此权重的计算与更新至关重要。考虑到当有检索系统中新的文档需要更新时,需要对权重进行自适应的更新,文献[6]提出了一种 SVD 重算的方法,在现有文档集合的基础上,对需要更新的文档建立新词关系矩阵,然后进行 SVD 重算。Bellegarda 等[7]提出了一种将新增加的文档建立的词关系矩阵代入(folding-in)原文档词关系矩阵的方法进行参数更新。但 SVD 重算方法与代入方法都存在更新时间开销大效率低的问题,无法适用于海量检索系统。文献[8]中提出了一种 SVD 更新的方法,该方法具有计算量少的特点,但 SVD 更新方法存在权重计算准确性低的问题,导致系统检索性能降低。

为了对 LSA 的权重函数进行有效的更新,提出了一种贝叶斯概率 LSA 模型权重自适应更新算法 ALSAB。ALSAB 采用最大后验概率估计与期望值最大(Expectation-Maximization,EM)算法[9]对 LSA 模型参数进行有效的估计,在充分考虑多次更新中不常用字词概率参数降低问题的前提下,采用增量学习方法降低多次更新产生的累积效应,提高检索的准确性与有效性。

基金项目:国家自然科学基金(the National Natural Science Foundation of China under Grant No.60842004);国家民委基金(the Natural Science Foundation of State Ethnic Affairs Commission under Grant No.08ZN02)。
作者简介:曾广平(1970-),男,讲师,主要研究领域为软件工程与信息检索。
收稿日期:2009-03-23 **修回日期:**2009-05-18

2　提出的算法

2.1　参数估计

考虑具有某种关联性的 K 个潜在主题 $z_k \in Z=\{z_1,\cdots z_k\}$,对于由文档与词组成对$(d_i,w_j)$的包含 N 个文档与 M 个词的集合 Y,有 $d_i \in \{d_1,\cdots d_N\}$,$w_j \in \{w_1,\cdots w_M\}$。根据 LSA 模型,词与文档在 K 个潜在主题上呈多项式分布,该分布概率用参数 $\theta=\{P(w_j|z_k),P(z_k|d_i)\}$表示。

根据最大后验概率估计,概率 LSA 模型参数 θ 可以采用后验概率 $P(\theta|X)$最大化的方法估计:

$$P(\theta|X)=\frac{P(X|\theta)}{P(X)}=\frac{\prod_{d_i\in D}\prod_{w_j\in W}P(d_i,w_j|\theta)P(\theta)}{P(X)} \tag{1}$$

其中,$P(\theta)$为先验概率,表示模型参数 $\theta=\{P(w_j|z_k),P(z_k|d_i)\}$的差异性,且满足:

$$\sum_{\forall w_j\in z_k}P(w_j|z_k)=1 \tag{2}$$

$$\sum_{\forall w_k\in d_i}P(z_k|d_i)=1 \tag{3}$$

假设随机变量 $P(w_j|z_k)$,$P(z_k|d_i)$独立,则先验概率 $P(\theta)$可以表示为:

$$P(\theta)\propto\prod_{k=1}^{K}\left[\prod_{j=1}^{M}P(w_j|z_k)^{\alpha_{j,k}-1}\prod_{i=1}^{N}P(z_k|d_i)^{\beta_{k,i}-1}\right] \tag{4}$$

其中 $\varphi=\{\alpha_{j,k},\beta_{k,i}\}$表示服从狄利克雷(Dirichlet)概率分布的参数,因此后验概率 $P(\theta|X)$可以表示为:

$$P(\theta|X)\propto\prod_{i=1}^{M}\prod_{j=1}^{N}P(w_j,d_i)^{n(w_j,d_i)}\cdot \prod_{k=1}^{K}\left[\prod_{j=1}^{M}P(w_j|z_k)^{\alpha_{j,k}-1}\prod_{i=1}^{N}P(z_k|d_i)^{\beta_{k,i}-1}\right] \tag{5}$$

表达式(1)中的后验概率等价于取对数后的相似函数 $P(\theta|X)$与先验概率 $P(\theta)$的对数和,因此有模型参数估计值 θ,采用对数最大似然估计有:

$$\theta=\arg\max_{\theta}P(\theta|X)=\arg\max_{\theta}\log P(X|\theta)+\log P(\theta) \tag{6}$$

由于表达式(6)中包含未知参数 z_k,采用 EM 算法进行估计,首先在 E 步中计算后验概率的期望值函数 $R(\hat{\theta}|\theta)$,对于表达式(5)有:

$$R(\hat{\theta}|\theta)\propto\prod_{j=1}^{M}\prod_{k=1}^{K}(\alpha_{j,k}-1)\log\hat{P}(w_j|z_k)+\prod_{i=1}^{N}\prod_{k=1}^{K}(\alpha_{j,k}-1)\log\hat{P}(z_k|d_i)+ \prod_{i=1}^{N}\prod_{j=1}^{M}n(d_i,w_j)\sum_{k=1}^{K}P(z_k|d_i,w_j)\log[\hat{P}(w_j|z_k)\hat{P}(z_k|d_i)]+ \eta_w\left(1-\sum_{j=1}^{M}\hat{P}(w_j|z_k)\right)+\eta_d\left(1-\sum_{j=1}^{M}\hat{P}(z_k|d_i)\right) \tag{7}$$

其中 η_w和 η_d分别为对该函数进行拉格朗日最优化求极值时的两个参数。

在 M 步中,先对期望函数 $R(\hat{\theta}|\theta)$分别对$\hat{P}(w_j|z_k)$和 η_w两个参数求偏导数,得:

$$\frac{R(\hat{\theta}|\theta)}{\partial\hat{P}(w_j|z_k)}=\frac{\sum_{i=1}^{N}n(d_i,w_j)P(z_k|d_i,w_j)+(r_{j,k}-1)}{\hat{P}(w_j|z_k)}-\eta_w=0 \tag{8}$$

$$\frac{R(\hat{\theta}|\theta)}{\partial\eta_w}=1-\sum_{j=1}^{M}\hat{P}(w_j|z_k)=0 \tag{9}$$

由式(8)和式(9)可计算出:

$$\hat{P}(w_j|z_k)=\frac{\sum_{i=1}^{N}n(d_i,w_j)P(z_k|d_i,w_j)+(r_{j,k}-1)}{\eta_w} \tag{10}$$

$$\eta_w=\sum_{j=1}^{M}\sum_{i=1}^{N}n(d_i,w_j)P_\theta(z_k|d_i,w_j)+(r_{j,k}-1) \tag{11}$$

由表达式(10)与(11)可解得$\hat{P}(w_j|z_k)$:

$$\hat{P}(w_j|z_k)=\frac{\sum_{i=1}^{N}n(d_i,w_j)P(z_k|d_i,w_j)+(\alpha_{j,k}-1)}{\sum_{m=1}^{M}[\sum_{i=1}^{N}n(d_i,w_j)P(z_k|d_i,w_j)+(\alpha_{j,k}-1)]} \tag{12}$$

同理,对期望函数 $R(\hat{\theta}|\theta)$分别对$\hat{P}(z_k|d_i)$和 η_d两个参数求偏导数,可解得$\hat{P}(z_k|d_i)$:

$$\hat{P}(z_k|d_i)=\frac{\sum_{i=1}^{N}n(d_i,w_j)P(z_k|d_i,w_j)+(\beta_{k,i}-1)}{n(d_i)+\sum_{l=1}^{K}(\beta_{l,i}-1)} \tag{13}$$

经过多次迭代运算,当到达终止条件时,可得概率 LSA 模型参数 θ 最大后验概率的估计参数集 $\theta=\{\hat{P}(w_j|z_k),\hat{P}(z_k|d_i)\}$。

2.2　增量学习

对于在线检索系统,由于文档集更新频繁,因此,概率 LSA 模型参数 θ 将会频繁更新,这将使得那些不常用的字词对应的概率参数减小,导致检索性能降低。为了降低这种累计效应,考虑新增加的 n 次文档集合 $C_n=\{X_1,\cdots,X_n\}$,则概率 LSA 模型参数的估计参数集 θ':

$$\theta'=\arg\max_{\theta}P(\theta|C_n)=\arg\max_{\theta}P(X_n|\theta)P(\theta|C_{n-1})\approx \arg\max_{\theta}P(X_n|\theta)P(\theta|\varphi_{n-1}) \tag{14}$$

其中,第 $n-1$ 次更新的后验概率 $P(\theta|C_{n-1})$将采用先验概率密度函数 $P(\theta|\varphi_{n-1})$近似计算,其中 φ_{n-1}由前 $n-1$ 次文档集合的累加计算:

$$\varphi_{n-1}=\{\alpha_{j,k}^{(n-1)},\beta_{k,i}^{(n-1)}\} \tag{15}$$

估计参数 θ'可表示为:

$$\theta'=\{P(w_j^{(n)}|z_k),P(z_k|d_i^{(n)})\} \tag{16}$$

后验概率的期望值函数 $R(\hat{\theta}'|\theta')$可表示为:

$$R(\hat{\theta}'|\theta')\propto\sum_{j=1}^{M}\sum_{k=1}^{K}[(\sum_{i=1}^{N}n(d_i^{(n)},w_j^{(n)})P^{(n)}(z_k|d_i^{(n)},w_j^{(n)})+ \alpha_{j,k}^{(n-1)}-1)\log\hat{P}^{(n)}(w_j^{(n)}|z_k)]+ \sum_{i=1}^{N}\sum_{k=1}^{K}[(\sum_{j=1}^{M}n(d_i^{(n)},w_j^{(n)})P^{(n)}(z_k|d_i^{(n)},w_j^{(n)})+ \beta_{k,i}^{(n-1)}-1)\log\hat{P}^{(n)}(z_k|d_j^{(n)})] \tag{17}$$

表达式(17)可以进一步表示为:

$$\exp(R(\hat{\theta}'|\theta'))\propto\sum_{k=1}^{K}\left[\sum_{j=1}^{M}\hat{P}^{(n)}(w_j^{(n)}|z_k)^{\alpha_{j,k}^{(n)}-1}\sum_{i=1}^{N}\hat{P}^{(n)}(z_k|d_i^{(n)})^{\beta_{k,i}^{(n)}-1}\right] \tag{18}$$

根据表达式(18)可得参数 $\alpha_{j,k}^{(n)}$与$\beta_{k,i}^{(n)}$的更新表达式:

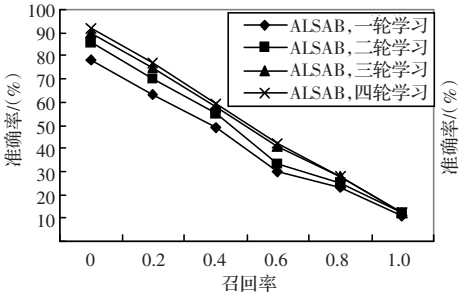

图1 学习周期对检索性能的影响

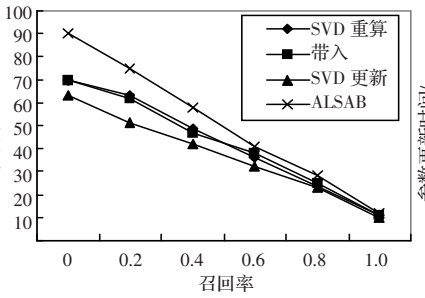

图2 四种算法检索准确率性能

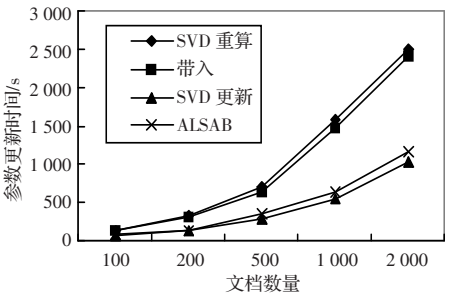

图3 四种算法参数更新时间性能

$$\alpha_{j,k}^{(n)}=\sum_{i=1}^{N}n(d_i^{(n)},w_j^{(n)})P^{(n)}(z_k|d_i^{(n)},w_j^{(n)})+\alpha_{j,k}^{(n-1)} \quad (19)$$

$$\beta_{k,i}^{(n)}=\sum_{j=1}^{M}n(d_i^{(n)},w_j^{(n)})P^{(n)}(z_k|d_i^{(n)},w_j^{(n)})+\beta_{k,i}^{(n-1)} \quad (20)$$

增量学习参数估计算法如下所示：

增量学习参数估计算法：

输入：训练文档集 Y，更新文档集 $C_n=\{X_1,\cdots,X_n\}$

输出：概率 LSA 模型参数的估计参数集 θ'

1.根据训练文档集 Y，估计初始参数 $\varphi_0=\{\alpha_{j,k}^{(0)},\beta_{k,i}^{(0)}\}$

2.Begin 自适应更新阶段

3.n=1

4. 输入更新文档集 X_n 与参数 φ_{n-1}

5. Begin 执行 EM 迭代算法估计参数

6. 根据每个单词 $w_j^{(n)}$ 与潜在变量 z_k 由表达式(12)计算 $\hat{P}(w_j|z_k)$

7. 根据每个文档 $d_i^{(n)}$ 与潜在变量 z_k 由表达式(13)计算 $\hat{P}(z_k|d_i)$

8. 当满足收敛条件时，结束 EM 迭代算法

9. END

10. 根据表达式(19)(20)更新参数 $\alpha_{j,k}^{(n)},\beta_{k,i}^{(n)}$

11. 输出概率 LSA 模型参数的估计参数集 $\theta'=\{P(w_j^{(n)}|z_k),P(z_k|d_i^{(n)})\}$

12. $n\leftarrow n+1$

13.END

3 实验结果

实验的硬件运行环境：CPU 为 Pentium 4、内存为512 MB、硬盘为 120 GB 的计算机；软件运行环境：Windows2000 Server 操作系统，VC++6.0；在 EM 算法的迭代停止阈值设为 0.01。实验数据采用 KDD 2005[9]中的两个数据集：BMS-WebView-1.dat 与 BMS-WebView-2.dat，其中 BMS-WebView-1.dat 包含 6 351 篇关于信息科学论文文档。问题产生方式由 BMS-WebView-1.dat 的收集者，邀请论文的作者根据论文研究的内容提出，经过筛选后产生 150 个查询句。BMS-WebView-2.dat 包含 3 678 篇关于医学科学论文文档，包含 50 个查询句。训练中取主题数为 10。为了验证提出的算法的有效性，将与文献[6-8]中提出的 SVD 重算、代入与 SVD 更新的算法进行对比，实验中以平均查准率与召回率评价算法的性能。

图 1 所示为学习周期对提出 ALSAB 算法性能的影响。如图 1 中所示，随着学习周期的增加，提出 ALSAB 算法的准确率将有所增加，但随着召回率的增加，准确率的增加将逐渐减少。经过三轮学习后，与一轮学习相比，准确率最多提高 12%，平均提高 8.7%。尽管经过第四轮学习后检索的准确率有所提高，但与三轮学习后的性能相差不大。因此考虑到学习所消耗的时间因素，后续试验中采用 3 轮学习训练。

图 2 所示为四种算法召回率与准确率性能。如图 2 中所示，由于 SVD 重算与代入两种算法采取的更新方式相似，两种算法的性能相当。SVD 更新算法由于只考虑部分更新文档的关系特征，因此存在权重计算准确性低的问题，导致系统检索性能降低，其性能最差。而提出的 ALSAB 算法性能明显优于其他算法，尤其是召回率较低时，准确率提高更明显。当召回率为 0.2 时，ALSAB 算法的准确率与 SVD 重算、代入、SVD 更新算法相比分别高 12%、13%与 24%；平均而言 ALSAB 算法的准确率分别高 8.5%、9.4%与 14.1%。其原因在于，提出的 ALSAB 算法采用最大后验概率估计与期望值最大(EM)算法对概率 LSA 模型参数进行有效的估计，而且充分考虑了多次更新中不常用字词概率参数降低问题，采用增量学习方法降低多次更新产生的累积效应，因而有效地提高了检索的准确率。

图 3 所示为四种算法参数估计与更新所需时间性能。如图 3 中所示，由于 SVD 重算与代入两种算法采取的更新方式相似，两种算法的参数更新时间性能相当。由于这两种算法需要完全重新计算，因此其参数更新时间随文档数量的增加而显著加大，当文档数量达到 2 000 时，参数更新时间分别达到 2 498 s 与 2 407 s，这表明这两种算法在时间开销上无法满足海量信息的更新与检索要求。SVD 更新算法由于只考虑部分更新文档的关系特征，因此其参数更新时间最少。而提出的 ALSAB 算法参数更新时间性能略低于 SVD 更新算法，且当更新文档数量较少时(如图 3 中所示低于 500 时)，两种算法的更新时间差并不明显。而且 ALSAB 算法参数更新时间明显优于 SVD 重算与代入两种算法，当文档数量为 2 000 个时，分别比 SVD 重算与代入算法少 1 333 s 与 1 242 s。这也表明了提出的 ALSAB 算法具有较好的参数更新时间性能。

4 结束语

针对潜在语义分析(LSA)模型的权重更新问题，提出了一种自适应权重更新算法 ALSAB。ALSAB 基于贝叶斯理论，采用最大后验概率估计与 EM 算法对参数进行有效的估计，并利用增量学习方法降低多次更新产生的累积效应。与现有的权重更新算法相比，实验结果表明提出的 ALSAB 算法不仅显著地提高了检索的准确率与召回率，而且具有较小的参数更新时间性能，能满足实际信息检索系统时间与性能上的要求。

参考文献：

[1] Wu Shengli.Applying statistical principles to data fusion in information retrieval[J].Expert Systems with Applications,2009,36(2):2997-3006.

(下转 102 页)