基于滑动窗口的主题模型

常东亚[1] 严建峰[1,2] 杨 璐[1] 刘晓升[1]
(苏州大学计算机科学与技术学院 苏州 215006)[1] (香港城市大学创意媒体学院 香港 999077)[2]

摘 要 LDA(Latent Dirichlet Allocation)是一个分层的概率主题模型,目前被广泛地应用于文本挖掘。这种模型既不考虑文档与文档之间的顺序关系,也不考虑同一篇文档中词与词之间的顺序关系,简化了问题的复杂性,同时也为模型的改进提供了契机。针对此问题提出了基于滑动窗口的主题模型,该模型的基本思想是文档中的一个单词的主题与其附近若干单词的主题关系越紧密,受附近单词主题的影响越大。根据窗口和滑动位移的大小,把文档切割为粒度更小的片段。同时,针对大数据集和数据流问题,提出了在线滑动窗口主题模型。在4个数据集上的实验表明,基于滑动窗口的主题模型训练出来的模型在数据集上有更好的泛化性能和精度。

关键词 潜在狄利克雷分配,主题模型,滑动窗口
中图法分类号 TP391 **文献标识码** A **DOI** 10.11896/j.issn.1002-137X.2016.12.018

Sliding-window Based Topic Modeling

CHANG Dong-ya[1] YAN Jian-feng[1,2] YANG Lu[1] LIU Xiao-sheng[1]
(School of Computer Science and Technology, Soochow University, Suzhou 215006, China)[1]
(School of Creative Media, City University of Hong Kong, Hong Kong 999077, China)[2]

Abstract LDA(Latent Dirichlet Allocation) is an important hierarchical Bayesian model for probabilistic topic modeling, which touches on many important applications of text mining. This model takes neither the order of documents nor the order of words in one document into account, which simplifies the complexity of issues and provides a great chance to improve itself. To achieve this goal, a sliding-window based topic model was proposed. The fundamental idea of this model is that the theme of one word in a specific document has a strong relationship at the words near by and is mainly affected by them. Through modifying the size of window and sliding step, document is cut into smaller pieces. Meanwhile, aiming at the big dataset and data flow, online sliding window theme model was proposed. Experiments show that the sliding-window based topic model has better generalization performance and accuracy on four common datasets.

Keywords Latentdirichlet allocation, Topic model, Sliding window

1 引言

随着信息技术的不断发展,网上的信息呈爆炸式增长,尤其是以文本形式为主要格式的数据更是以指数的形式增长。面对如此庞大的信息数据,要想从中挑选出需要的信息是一件非常复杂的事情。正是由于这样的问题促使了主题模型的产生。潜在狄利克雷(LDA)[1]是当前非常流行的一种文档主题生成模型。它是一个3层贝叶斯概率模型(HBM),包含词、主题、文档3层结构。其自2003年Blei等人[2]提出以后,在文本挖掘、计算机视觉和计算生物学等领域都已经得到了很好的应用。LDA被认为是分析大规模的非结构化文档集合的最有效的工具。因为在其图模型中有许多循环,目前,LDA还没有一个精确的推理方法,最常用的近似推理算法有塌陷吉布斯采样(Gibbs Sampling,GS)[3]、变分贝叶斯(Variational Bayes,VB)[1]、消息传递(Belief Propagation,BP)[4-6,19],以及在其基础上的扩展算法,包括作者主题模型[7]、相关主题模型[8]等。另外还有一些其他的主题模型推理方法,包括期望传播(Expectation Propagation)[9]、塌陷变分推理(CVB)[10]等。文献[11]对这些方法进行了理论与实验证明。针对大数据流语料库,最常用的算法有在线吉布斯采样(OGS)[12]、在线变分贝叶斯(OVB)[13]以及在线消息传递(OBP)[14,15]等。针对离线数据集,本文提出了一种基于滑动窗口的主题模型(Sliding Window Topic Mo-del,SWTM),并与吉布斯采样(GS)、变分贝叶斯(VB)和消息传递(BP)做了实验对比。基于在线数据流,本文提出了一种基于滑动窗口的在线滑动窗口主题模型(Online Sliding Window Topic Model,OSWTM),并与在线吉布斯采样(OGS)、在线变分推理(OVB)做了实验对比。实验表明,基于滑动窗口的主题模型和在线滑动窗口主题模型在未知数据集上具有更好的泛化能力和精度。

本文第2节简要地介绍了LDA模型和推理,以及一些常

到稿日期:2015-11-26 返修日期:2016-03-08 本文受国家自然科学基金(61373092,61572339,61272449),江苏省科技支撑计划重点项目(BE2014005)资助。

常东亚(1992-),男,硕士生,主要研究方向为机器学习,E-mail:20144227043@stu.suda.edu.cn;严建峰(1978-),男,副教授,硕士生导师,主要研究方向为机器学习、传感器网络;刘晓升(1976-),男,博士生,主要研究方向为机器学习。

用的 LDA 近似推理算法；第 3 节提出了基于滑动窗口的主题模型(SWTM)，以及 SWTM 的图模型和推理；第 4 节主要介绍离线的基于滑动窗口的主题模型，并在 4 个数据集上与吉布斯采样(GS)、变分推理(VB)和消息传递(BP)做了实验对比；第 5 节介绍了基于滑动窗口的在线主题模型以及推理，并在 4 个数据集上与在线吉布斯采样(OGS)、在线变分推理(OVB)做了实验对比；最后总结全文。

2 相关工作

2.1 潜在狄利克雷分配(LDA)

潜在狄利克雷分配(LDA)是一个文档生成模型，它是一个 3 层贝叶斯模型(HBM)，包含词、主题、文档 3 层结构。LDA 模型是一个基于词袋(Bag of Words，BOW)的模型，不考虑文档与文档之间的顺序，同时也不考虑每篇文档内单词与单词之间的顺序。图 1 是 LDA 的图模型，图中的阴影圆圈表示可观测变量，非阴影圆圈表示潜在变量，箭头表示两个变量之间的条件依赖性，方框表示重复采样，方框右下角的数字表示重复采样的次数。LDA 假设一篇文档是主题上的分布，而每一个主题又是单词表上单词的分布。LDA 的建模过程是逆向通过文本集合建立生成模型，一篇文档的生成过程按照以下 4 个步骤进行：

(1)首先根据文档变量 θ_d 满足的先验分布 $\theta_d \sim Dir(\alpha)$，随机选择一个文档-主题分布 θ_d，其中 $1 \leqslant d \leqslant D$。

(2)然后根据单词变量 ϕ_k 满足的先验分布 $\phi_k \sim Dir(\beta)$，随机选择一个主题-单词分布 ϕ_k，其中 $1 \leqslant k \leqslant K$。

(3)对文档 d 中的每个单词 w，首先根据文档-主题分布选择一个主题 z_k，其中 $z_k \sim Mult(\theta_d)$，然后根据主题-单词分布选择一个单词 w，其中 $w \sim Mult(\phi_{z_K})$。

(4)不断循环步骤(1)－(3)，直至生成整个文档集合。由此可以得到一篇文档的联合概率分布为：

$$P(Z,\theta,\phi)=\frac{P(\theta,\phi,Z,W|\alpha,\beta)}{P(D|\alpha,\beta)} \tag{1}$$

其中，分子为随机变量的联合分布，对于任意值的隐藏变量，容易求得它的联合概率分布，而分母计算量十分庞大，不容易求解，因此通常采用一些近似推理算法来求解。常用的近似推理算法有变分推理(VB)、吉布斯采样(GS)以及消息传递(BP)。吉布斯采样是基于马尔可夫链蒙特卡尔理论(MCMC)采样，从后验分布中为每个单词采样一个主题，马尔科夫链收敛后的稳定分布就是后验分布。变分推理是一个优化方法，根据 KL 散度把后验分布近似简化到一个易于求解的分布。因为近似简化后有一定的偏差，在实验中吉布斯采样方法比变分推理表现得更好，但是在大数据集中，变分推理的精度可能更高[16,17]。

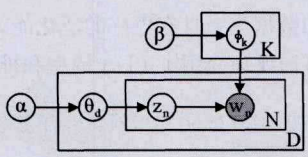

图 1 LDA 图模型

表 1 列出了本文使用到的一些标记符号。

表 1 符号标签

符号	意义
$1 \leqslant d \leqslant D$	语料库文本索引
$1 \leqslant w \leqslant W$	单词表中单词索引
$1 \leqslant k \leqslant K$	主题索引
$x_{w,d}$	索引为{w,d}的单词的数目
θ_d	文档 d 的主题分布
ϕ_k	主题 k 对应的单词分布
z_k	主题 k
$z_{-w,d}$	文本 d 中除 w 外所属的主题
$z_{w,-d}$	单词 w 除文本 d 外所属的主题
$\mu(z_{w,d}=k)$	文档 d 中单词 w 分配主题 k 的概率
$\mu(z_{.,d})$	$\sum_w \mu(z_{w,d})$
$\mu(z_{w,.})$	$\sum_d \mu(z_{w,d})$
$\theta_{k\|d}$	文本 d 在主题上的分布
$\phi_{w\|k}$	单词 w 的因子
$n_{k\|d}$	被分配给主题 k 的文档 d 的个数
$n_{w\|k}$	被分配给主题 k 的单词 w 的个数
α,β	狄利克雷超参
N_d	文档 d 划分的窗口数
$win_{n,d}$	文档 d 中第 n 个窗口
$w^d_{win,i}$	文档 d 中窗口 win 的第 i 个单词
$Z^d_{win,i}$	文档 d 中窗口 win 第 i 个单词的主题标签

LDA 模型的目标是在给定的文档数据集 D 的条件下，推断出文档对应的主题分布 θ_d、主题对应的单词分布 ϕ_k 以及隐藏主题的概率分布，并使用后验概率来推理计算这些参数。

2.2 BP 算法

文献[4]首次使用消息传递(BP)算法来求解 LDA 模型。消息传递算法是将基于 LDA 模型的概率图表示转化成等价的因子图，如图 2 所示。

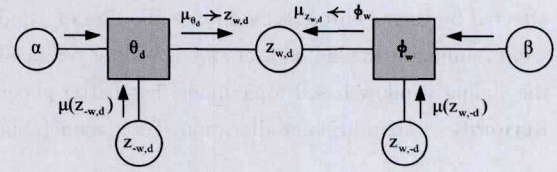

图 2 BP 算法的因子图

其中，主题标签 $z_{w,d}$ 的邻居分别是 $z_{w,-d}$ 和 $z_{-w,d}$，标签 $z_{-w,d}$ 表示文档 d 中除单词 w 外的所有单词索引的主题标签，$z_{w,-d}$ 表示单词 w 在除文档 d 外的所有文档的主题标签。LDA 模型中消息传递算法并不直接计算后验分布 $p(z|w)$，而是计算边缘概率 $p(z_{w,d})$，也称为消息 $\mu(z_{w,d})$。如图 2 所示，一篇文档 d 的一个单词 w 不仅受到不同文档中的同一个单词影响，记为 $x_{w,-d} * \mu(z_{w,-d})$，还受到同一篇文档中不同单词的影响，记为 $x_{-w,d} * \mu(z_{-w,d})$。因此消息 $\mu_{w,d}$ 等于其从邻居因子获取的消息之积，即：

$$\mu(z_{w,d}) \propto \mu_{\theta_d \to z_{w,d}}(z_{w,d}) \times \mu_{\phi_w \to z_{w,d}}(z_{w,d}) \tag{2}$$

消息更新公式为：

$$\mu(z_{w,d}=k) \propto \frac{\tilde{\mu}(z_{-w,d}=k)+\alpha}{\sum_k [\tilde{\mu}(z_{-w,d}=k)+\alpha]} \times \frac{\tilde{\mu}(z_{w,-d}=k)+\beta}{\sum_w [\tilde{\mu}(z_{w,-d}=k)+\beta]} \tag{3}$$

其中，

$$\tilde{\mu}(z_{-w,d}=k)=\sum_{-w} x_{-w,d}\mu(z_{-w,d}=k) \tag{4}$$

$$\tilde{\mu}(z_{w,-d}=k)=\sum_{-d} x_{w,-d}\mu(z_{w,-d}=k) \tag{5}$$

由此得到文档主题分布和主题单词分布的更新公式为：

$$\theta_{k|d}=\frac{\tilde{\mu}(z_{.,d}=k)+\alpha}{\sum_{k}[\tilde{\mu}(z_{.,d}=k)+\alpha]} \tag{6}$$

$$\phi_{w|k}=\frac{\tilde{\mu}(z_{w,.}=k)+\beta}{\sum_{w}[\tilde{\mu}(z_{w,.}=k)+\beta]} \tag{7}$$

3 SWTM

潜在狄利克雷(LDA)模型采用一个基于词袋(Bag of Words)的方法，这种方法将每一篇文档视为一个词频向量，从而将文本信息转化为易于建模的数字信息，但是这种基于词袋的方法把每一篇文档作为一个整体词袋，没有考虑文档与文档之间的顺序关系，也没有考虑每篇文档中词与词之间的顺序关系，其基本思想是一篇文档中的一个单词的主题标签不仅受到同一篇文档中不同单词主题标签的影响，还受到不同文档中同一个单词主题标签的影响，这简化了问题的复杂性，同时也为模型的改进提供了契机。因此提出了基于滑动窗口的主题模型，图 3 是滑动窗口的图模型，模型的基本思想是一篇文档中一个单词的主题与附近几句话或者几个段落的单词主题的相关性最大。每篇文档根据窗口(Win)和滑动位移(Slide)的大小将文档有交叉重叠的划分为更小粒度上的片段，训练时，每个窗口内的单词是无序的，但是由于窗口的滑动，造成了每篇文档不同窗口之间是局部有序的。这样划分的依据是在一篇文档中，每个单词的主题和这个单词所在的附近的几个单词或者几句话的主题关系更为紧密。在滑动窗口主题模型中(SWTM)，一个窗口中单词的主题标签除了受到同一窗口中不同单词主题的影响外，还受到不同文档中同一个单词主题的影响，以及同一文档中，除了窗口中的单词外，同一个单词主题的影响。

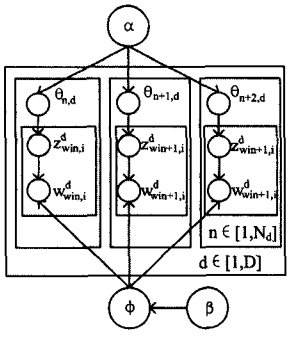

图 3 SWTM 图模型

根据划分窗口时文档衔接处的不同处理方式，又可以分为两种不同的情况：有缝划分和无缝划分。有缝划分是指把一篇文档划分为若干片段，但是相邻的文档之间单词不会划分到同一个窗口之中(例如当文档 d 的末尾不够一个窗口大小时，将其当作一个窗口)；无缝划分是指在一篇文档的末尾不够一个窗口的大小时，用下一篇文档的前几个单词来补上，凑够一个窗口的大小。但是当最后一篇文档的末尾不够一个窗口大小时，将其当作一个窗口来计算。本文实验中采取的是有缝划分方法。

根据图 3 所示的基于滑动窗口的主题模型(SWTM)的图模型，文档生成过程如图 4 所示。

SWTM 生成过程：

1. For 对每一个主题 k∈[1,k] do
2. 采样一个主题单词分布 $\phi_k\sim Dir(\beta)$
3. End For
4. For 对每一篇文档 d∈[1,D] do
5. For 对每一个窗口 $win_{n,d}\in[1,N_d]$ do
6. 采样一个窗口主题分布 $\theta_{n,d}\sim Dir(\alpha)$
7. For 对窗口中的每个词 $w^d_{win,i}\in[1,N_{win}]$ do
8. 采样一个主题 $z^d_{win,i}\sim Mult(\theta_{n,d})$
9. 采样一个单词 $w^d_{win,i}\sim Mult(\phi_{z^d_{win,i}})$
10. End for
11. End for
12. End for

图 4 SWTM 生成过程

一篇文档 d 中的一个单词 w 受到不同窗口中相同单词影响，记为 $x_{w,-win}*\mu(z_{w,-win})$，以及同一个窗口中不同单词的影响，记为 $x_{-w,win}*\mu(z_{-w,win})$。消息 $\mu_{w,win}$ 的更新公式为：

$$\mu_{w,win}(k)\propto\frac{[\hat{\theta}_{-w,win}(k)+\alpha][\hat{\phi}_{w,-win}(k)+\beta]}{\hat{\phi}_{-(w,win)}(k)+W\beta} \tag{8}$$

其中：

$$\hat{\theta}_{-w,win}(k)=\sum_{-w}x_{w,win}*\mu_{w,win}(k) \tag{9}$$

$$\hat{\phi}_{w,-win}(k)=\sum_{-win}x_{w,win}*\mu_{w,win}(k) \tag{10}$$

通过归一化得到：

$$\theta_d(k)=\frac{\sum_{win}\hat{\theta}^d_{win}(k)+\alpha}{\sum_{K}\sum_{win}\hat{\theta}^d_{win}(k)+K\alpha} \tag{11}$$

$$\phi_w(k)=\frac{\hat{\phi}_w(k)+\beta}{\sum_{w}\hat{\phi}_w(k)+W\beta} \tag{12}$$

根据上面的更新公式，得到 SWTM 训练过程如图 5 所示。

SWTM 的训练过程如下：

首先用滑动窗口把数据集划分为有交叠的 N 个窗口{win_i}，1≤i≤N，然后通过以下过程进行训练：

1. 输入参数：α，β，D，K
2. 输出：$\theta_{k\times N}$，$\phi_{k\times w}$
3. 初始化和归一化 $\mu_{w,win}(k)$
4. Repeat
5. 遍历语料库，对每个窗口中的每个单词用式(8)计算维度 K 的主题分布，并保留这 K 个信息量更新；
6. 根据 $\mu_{w,win}$ 按照式(9)和式(10)来更新 $\hat{\theta}_{-w,win}(k)$，$\hat{\phi}_{w,-win}(k)$
7. Until 收敛或者达到迭代次数
8. 用 $\hat{\theta}_{-w,win}(k)$，$\hat{\phi}_{w,-win}(k)$ 按照式(11)和式(12)来计算模型参数 $\theta_{k\times N}$，$\phi_{k\times w}$；
9. 输出参数 $\theta_{k\times N}$，$\phi_{k\times w}$

图 5 SWTM 的训练过程

4 SWTM 实验分析

4.1 实验环境和数据集

本文实验是基于单机多核服务器进行的，该服务器有 2 个 Intel Xeon X5690 3.47G 的 CPU，每个 CPU 有 6 个核，总计 12 个核，140GB 内存。

实验在 4 个数据集上进行，分别为 20newsgroup(20ng)、

Reuters 8(R8)、Reuters 52(R52)和 Webkb,这 4 个数据集公布在 Dr. SELIM MIMAROGLU[1)] 的主页上,4 个数据集情况的说明如表 2 所列。

表 2 数据集统计

数据集	Train	Test	W	Stop
20ng	11293	7528	93864	no
R8	5485	2189	23585	no
R52	6532	2568	26284	no
Webkb	2785	1396	7770	no

表 2 概括统计了实验所用的 4 个数据集,*Train* 是训练集文档数,*Test* 是测试集文档数,W 是单词表的大小,*Stop* 指是否含有停用词,*no* 指数据集不含有停用词。

在许多关于主题模型的实验以及应用中,先验参数选取对于模型的好坏有很大的影响,由于先验参数的选取并不是本文研究重点,因此省略先验参数的推理。为了简化实验以及算法的公平比较,实验中全部使用对称固定的先验参数 $\alpha=0.01$,$\beta=0.01$,关于更多的先验参数的研究可以参考文献[18]。

4.2 评价标准

主题模型被提出之后,需要对模型的好坏进行评估,依次判断改进的参数或者算法的建模能力是否得到提高。常见的评价聚类效果的方法在训练模型中不使用训练数据的类别去计算数据的对数似然 $\log p(W_d \mid D)$,因为对数似然值通常是一个很大的负数,因此最初在主题模型中使用的混淆度(perplexity)经常被用来评估主题模型的性能。混淆度是一种信息理论的测量方法,经常被用在统计语言模型中衡量语言模型对测试语料库建模能力的好坏。

混淆度的计算公式为:

$$perplexity=\exp\{-\frac{\sum_{w,d} x_{w,d} * \log[\sum_{k}\theta_d(k)\phi_w(k)]}{\sum_{w,d} x_{w,d}}\}$$

混淆度是每个单词似然几何均值的相反数,当新的模型算法被提出之后,通常与标准算法在测试集上进行混淆度的对比,混淆度值越低表示模型的泛化能力越好。

4.3 算法自身性能分析

SWTM 算法引入了两个参数:*Win* 和 *Slide*,分别表示窗口的大小和滑动位移的大小。算法的性能与这两个参数的取值有很大的关联,因此在进行算法性能对比实验之前,需要先做一些实验来确定 SWTM 算法在 4 个数据集上与 *Win* 和 *Slide* 的关系(20ng * 0.3 表示在 20ng 数据集上的真实实验结果乘以 0.3,以便在图中与其它数据集上的结果更好地呈现在同一张图表中)。

图 6 示出了在 4 个数据集上混淆度随滑动位移变化的情况,在实验中,主题数设为 $K=100$,窗口的大小设为 $Win=100$。从实验结果可知,在窗口长度范围内,滑动位移越大,混淆度越小,模型的性能也越好。当滑动位移的大小越接近窗口大小时,模型的混淆度也趋于稳定值。当滑动位移 $Slide=Win$ 时,混淆度最小,这时是无交叉划分,相邻的两个窗口之间没有交叉重叠。

图 7 是在 4 个数据集上混淆度随窗口大小变化的情况,滑动位移的大小设为 $Slide=30$,主题数 $K=100$。从实验结果可以看出,窗口越小,混淆度越小,训练得到的模型泛化能力越好,这也说明了把文档切分为更小粒度的片段的必要性。窗口增大到一定大小时,模型的混淆度趋于稳定。

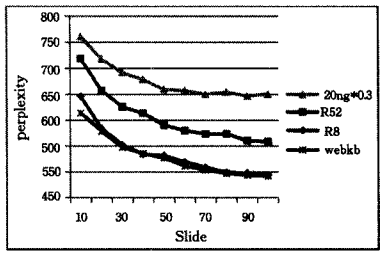

图 6 混淆度随滑动位移变化的情况

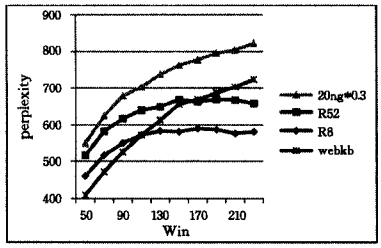

图 7 混淆度随窗口大小变化的情况

4.4 实验对比分析

图 8—图 11 分别是 SWTM 与吉布斯采样(GS)[3]、变分推理(VB)[1]、消息传递(BP)[4]在 4 个数据集上的实验结果对比分析。实验中主题数 $K=\{20,40,60,80,100,120,140,160,180,200\}$,SWTM 的滑动窗口 $Win=40$,滑动位移 $Slide=20$。从实验结果可以看出,基于 SWTM 训练出来的模型在数据集上的预测混淆度是最小的,而混淆度反映的是当前训练模型对未知数据的预测能力,混淆度的值越小说明模型的预测准确性越好,因此 SWTM 训练出来的模型的预测能力最好。从实验结果还可以看出,在数据集很小的情况下,SWTM 训练出来模型的预测准确性比其它算法要好很多,这说明 SWTM 在小数据集上也会有很好的预测性能。

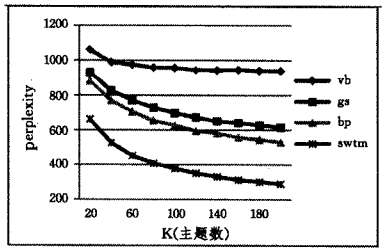

图 8 webkb 数据集上混淆度的比较

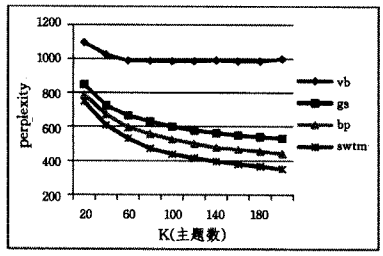

图 9 R8 数据集上混淆度的比较

1) http://www.cs.umb.edu/~smimarog/textmining/datasets

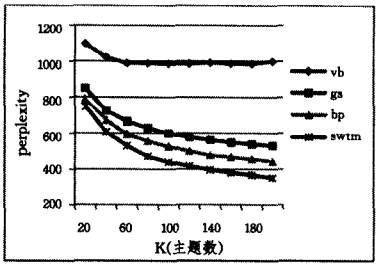

图 10　R52 数据集上混淆度的比较

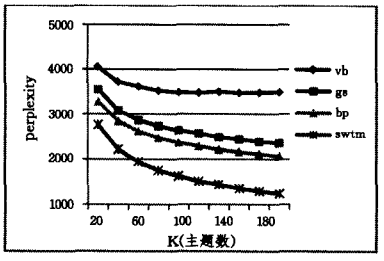

图 11　20ng 数据集上混淆度的比较

图 12 和图 13 是 4 种算法的时间分析，实验中主题数分别设为 $K=100$ 和 $K=200$，迭代次数设为 500，这足以使每个算法都达到收敛。从实验结果可以看出，SWTM 算法的运行时间比 GS 和 BP 算法多一点，这是由于把文档划分为粒度更小的片段，并且每个片段之间有重叠的结果。从中可以看出 VB 算法的运行时间最长，这是由于在 VB 算法中计算 digamma 函数需要花费大量的时间，造成 VB 算法每次迭代时都要花费大量时间。但 VB 算法是最快达到收敛的，一般不会超过 100 次迭代(VB * 0.2 表示 VB 算法的实际运行时间乘以 0.2 倍，以便更好地在图中与其它算法作对比)。

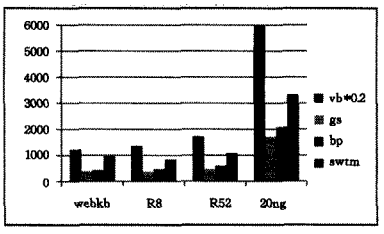

图 12　$K=100$ 时算法运行时间比较

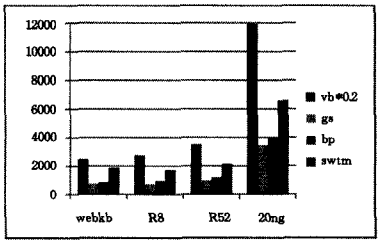

图 13　$K=200$ 时算法运行时间比较

5　在线 SWTM

离线算法是假设训练数据在处理前能够完整获取，并且保存在内存里，因此在离线数据集上，VB，GS，BP 以及本文提出的 SWTM 都能很好地工作，然而实际中数据量会随着时间不断地增加，形成一个数据流，此时这些离线算法由于内存和算法本身的限制而不能很好地工作。

目前，在主题模型领域已经提出了很多在线学习算法，例如在线 GS(OGS)[12]、在线 VB(OVB)[13]。这些在线学习算法分别是在对应的基本算法 GS、VB 的基础上提出来的。因此本文在 SWTM 基础上提出了一种在线学习算法，即在线滑动窗口主题模型(OSWTM)。

在线滑动窗口主题模型的基本思想是首先把庞大的数据集分割成小的数据块 D_s，然后在每个小数据块里用 SWTM 算法进行训练。训练第一个数据块和离线的 SWTM 是一样的，保存训练结束后的 $\phi_w(k)$，从第二个数据块开始，一直到最后一段，只需要初始化 $\theta_d(k)$，而 $\phi_w^s(k)$ 的初始值为上一段保留下来的 $\phi_w^{s-1}(k)$，一直不停迭代直到收敛或者达到停止条件，最后得到一个新的 $\phi_w(k)^{new}$。图 14 是在线滑动窗口主题模型的训练过程。

OSWTM 训练过程：

首先把数据集划分为若干个数据块 D_S，每个数据块包含 M 篇文档，然后使用滑动窗口把每个小数据块分割成交叠的 N 个窗口 $\{win_i\}$，$1\leqslant i\leqslant N$。然后通过下面的过程进行训练。

1. 输入：$x^s_{w,win}, D_s, K, \alpha, \beta$
2. 输出：$\phi_{K\times W}, \theta_{K\times N}$
3. For 对第一块数据
4. 　　初始和归一化 $\mu^0_{w,win}(k)$
5. 　　$\phi^1_{K\times W}=SWTM(x^1_{w,win}, \phi^0_{K\times W}, K, \alpha, \beta)$，保存并释放内存
6. End For
7. For S=2:N 块数据
8. 　　把 $x^s_{w,win}$ 和 $\phi^{s-1}_{K\times W}$ 加载到内存
9. 　　$\phi^s_{K\times W}=SWTM(x^s_{w,win}, \phi^{s-1}_{K\times W}, K, \alpha, \beta)$，保存并释放内存
10. End For

图 14　OSWTM 训练过程

5.1　算法自身性能分析

在线滑动窗口主题模型(OSWTM)引入了 3 个参数：窗口 *Win*、滑动位移 *Slide*、数据块大小 *M*。模型的性能与 3 个参数有关。图 15—图 17 展示了在 4 个数据集上模型的性能与 3 个参数之间的关系，实验中主题个数设为 $K=100$。

图 15 示出了混淆度随滑动位移的变化情况。实验中，窗口 $Win=100$，由于 webkb 数据集比较小，每个数据块的大小设为 $M=200$，其它 3 个数据集数据块大小设为 $M=500$，滑动位移 $Slide=\{10,20,30,40,50,60,70,80,90,100\}$。从实验结果可以看出，混淆度随滑动位移的增大而减小，但是当滑动位移增大到一定程度时，混淆度没有很大的变化，基本趋于稳定。例如当 $Slide=50$ 时，混淆度基本趋于一个稳定值。

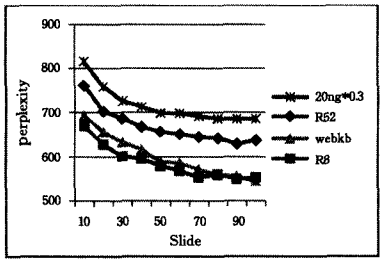

图 15　混淆度随滑动位移的变化情况

图 16 是混淆度随窗口的变化情况。实验中，滑动位移 $Slide=20$，webkb 数据集中每块文档数设为 $M=200$，其它 3 个数据块的大小设为 $M=500$，窗口 $Win=\{50,70,90,110,130,150,170,190,210,230\}$。从实验结果可以看出，当窗口

越小时混淆度也越小，训练出来的模型泛化能力越好，当窗口增大到一定程度时，混淆度也趋于稳定，不再变化。

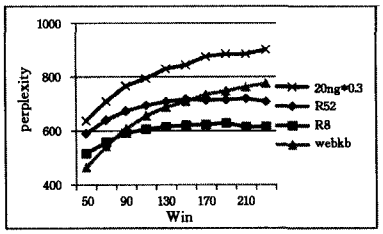

图 16　混淆度随窗口的变化情况

图 17 是混淆度与每个数据块文档数之间的关系。实验中，窗口 $Win=40$，$Slide=20$，每个数据块的文档数 $M=\{50, 100, 150, 200, 250, 300, 350, 400, 450, 500\}$。从实验结果可以看出，随着文档数的增多，混淆度虽然有所减小，但是减小得不是很明显，说明模型与文档块的大小没有很紧密的关系，因此在实际应用中可以将很大的数据集划分为更小的数据集，模型的泛化能力和精度不减。

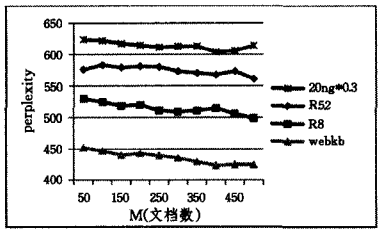

图 17　混淆度随每个数据块的大小变化情况

5.2　实验对比

图 18－图 21 给出了主题模型在 20ng、R8、R52、webkb 4 个数据集上分别采用 OSWTM，OGS[12] 以及 OVB[13] 算法的预测混淆度的对比。其中混淆度越小说明模型对未知数据的预测能力越好。为了对比的公平性，统一设定数据块中的文档数 $M=500$，从实验结果可以看出，OSWTM 在 4 个数据集上的混淆度均低于另外 3 个算法。这表明在主题模型上 OSWTM 算法相比其它 3 个算法得到的训练结果更为精确，在大数据和数据流方面训练出来的模型也有很好的泛化能力。

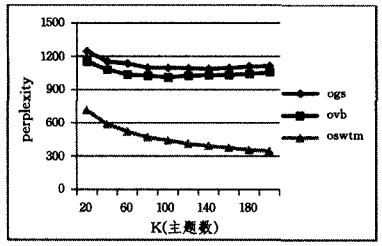

图 18　webkb 数据集上算法的比较

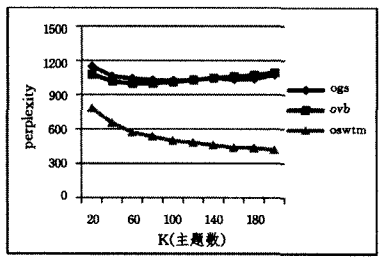

图 19　R8 数据集上算法的比较

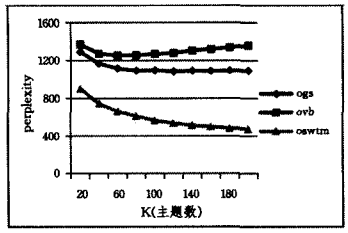

图 20　R52 数据集上算法的比较

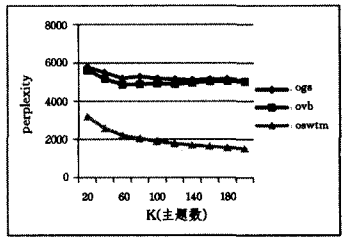

图 21　20ng 数据集上算法的比较

结束语　本文在 LDA 的基础上提出了基于滑动窗口的主题模型(SWTM)，并与目前最流行的主题模型吉布斯采样(GS)、变分推理(VB)和置信传播(BP)进行了实验对比。通过实验对比，本文提出的基于滑动窗口的(SWTM))主题模型在数据集上有更好的泛化能力和精度。同时，通过实验比较，本文提出的在线滑动窗口主题模型在大数据流上，比在线 GS(OGS)和在线 VB(OVB)有更好的表现。但是基于滑动窗口的主题模型在实际应用以及并行运行方面的性能还有待研究，这是一个值得研究的课题，同时也是我们下一步的研究方向。

参 考 文 献

[1] Blei D M, Ng A Y, Jordan M I. Latent Dirichlet allocation[J]. J. Mach. Learn. Res., 2003(3): 993-1022

[2] Blei D M. Introduction to Probabilistic Topic Models[J]. Communications of the ACM, 2011, 27(6): 55-65

[3] Griffiths T L, Steyvers M. Finding scientific topics [J]. Proceedings of the National Academy of Sciences, 2004, 101(Suppl 1): 5228-5235

[4] Zeng J, Cheung W K, Liu J. Learning Topic Models by Belief Propagation[J]. IEEE Transactions on Pattern Analysis & Machine Intelligence, 2013, 35(5): 1121-1134

[5] Wu X, Zeng J, et al. Finding Better Topics: Features, Priors and Constraints[M]// Advances in Knowledge Discovery and Data Mining. Springer International Publishing, 2014: 296-310

[6] Zeng J. A topic modeling toolbox using belief propagation[J]. The Journal of Machine Learning Research, 2012, 13(1): 2233-2236

[7] Rosen-Zvi M, Griffiths T, Steyvers M, et al. Theauthor-topic model for authors and documents[C]//UAI. 2004: 487-494

[8] Chang J, Blei D M. Hierarchical Relational models for Document Networks[J]. EprintArxiv, 2009, 4(1): 124-150

[9] Takita M, Naziruddin B, Matsumoto S, et al. Expectation-Propogation for the Generative Aspect Model[J]. Computer Science, 2002, 235(11): 3257-3269

[10] Schölkopf B, Platt J, Hofmann T. A Collapsed Variational Bayesian Inference Algorithm for Latent Dirichlet Allocation[J]. Advances in Neural Information Processing Systems, 2006(19): 1353-1360

[11] Asuncion A, Welling M, Smyth P, et al. On smoothing and inference for topic models[C]// Proceedings of the Twenty-Fifth

Conference on Uncertainty in Artificial Intelligence. AUAI Press,2009:27-34

[12] Yao L, Mimno D, McCallum A. Efficient methods for topic model inference on streaming document collections[C]// Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. ACM,2009:937-946

[13] Hoffman M, Bach F R, Blei D M. Online learning for latent dirichlet allocation[C]// Advances in Neural Information Processing Systems. 2010:856-864

[14] Zeng J, Liu Z Q, Cao X Q. Fast Online EM for Big Topic Modding[J]. IEEE Transactions on Knowledge & Data Engineering,2016,28(3):675-688

[15] Ye Y, Gong S, Liu C, et al. Online belief propagation algorithm for probabilistic latent semantic analysis[J]. Frontiers of Computer Science,2013,7(4):526-535

[16] Asuncion A, Welling M, Smyth P, et al. On smoothing and inference for topic models[C]// Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence. AUAI Press,2009:27-34

[17] Braun M, McAuliffe J. Variational inference for large-scale models of discrete choice[J]. Journal of the American Statistical Association,2010,105(489):324-335

[18] Wallach H M, Mimno D M, Mccallum A. Rethinking LDA: why priors matter[J]. Advances in Neural Information Processing Systems,2009(23):1973-1981

[19] Gao Yang, Yang Lu, Liu Xiao-sheng, et al. Study of Semantic Understanding by LDA[J]. Computer Science,2015,42(8):279-282(in Chinese)
高阳,杨璐,刘晓升,等. LDA 语义理解研究[J]. 计算机科学,2015,42(8):279-282

(上接第 83 页)

[5] Zhang Ying-chun, Su Bo-hong, Cao Juan. Study on application of attributive reduction based on Rough sets in Data Mining[J]. Computer Science,2013,40(8):223-226(in Chinese)
张颖淳,苏伯洪,曹娟. 基于粗糙集的属性约简在数据挖掘中的应用研[J]. 计算机科学,2013,40(8):223-226

[6] Li Ming, Deng Shao-bo, Feng Sheng-zhong, et al. Fast assignment reduction in inconsistent incomplete decision systems[J]. Journal of Systems Engineering & Electronics,2014,25(1):83-94

[7] Zhang Qin-hua, Guo Yong-hong, Xiao Yu. Attribute Reduction Based on Approximation Set of Rough Set[J]. Journal of Computational Information Systems,2014,10(16):6859-6866

[8] Wang Yong-sheng, Zheng Xue-feng, Suo Yan-feng. Dynamic algorithm for computer attribute reduction based on information granularity[J]. Computer Science,2015,42(4):213-216(in Chinese)
王永生,郑雪峰,锁延锋. 一种基于信息粒度的动态属性约简求解算法[J]. 计算机科学,2015,42(4):213-216

[9] Yang Xi-bei, Yan Xu, Xu Su-ping, et al. New Heuristic Attribute Reduction Algorithm Based on Sample Selection[J]. Computer Science,2016,43(1):49-52(in Chinese)
杨习贝,颜旭,徐苏平,等. 基于样本选择的启发式属性约简方法研究[J]. 计算机科学,2016,43(1):49-52

[10] R I, CT G, Enriquez S, et al. Attributing reductions in coral calcification to the saturation state of aragonite, comments on the effects of persistent natural acidification[J]. Proceedings of the National Academy of Sciences of the United States of America,2014,111(3):E300-E301

[11] Wang Chang-zhong, He Qiang, Chen De-gang, et al. A novel method for attribute reduction of covering decision systems[J]. Information Sciences,2014,254(5):181-196

[12] Qian Jin, Lv Ping, Yue Xiao-dong, et al. Hierarchical attribute reduction algorithms for big data using MapReduce[J]. Knowledge-Based Systems,2015,73(12):18-31

[13] Skowron A, Rauszer C. The Discernibility Matrices and Functions in Information Systems[J]. Theory & Decision Library,2012,11:331-362

[14] Hu Xiao-hua, Cercone N. Learning in Relational Data-bases: A Rough Set Approach[J]. Computational Intelligence, 1995, 11(2):323-338

[15] Ye Dong-yi, Chen Zhao-jiong. A new discernibility matrix and the computation of a core[J]. Chinese Journal of Electronic,2002,30(7):1086-1088(in Chinese)
叶东毅,陈昭炯. 一个新的差别矩阵及其求核方法[J]. 电子学报,2002,30(7):1086-1088

[16] Wang Guo-yin. Calculation methods for core attributes of decision table [J]. Chinese Journal of Computers,2003,26(5):611-615(in Chinese)
王国胤. 决策表核属性的计算方法[J]. 计算机学报,2003,26(5):611-615

[17] Zhao Jun, Wang Guo-yin, Wu Zhong-fu, et al. An efficient approach to compute the feature core[J]. Mini-Micro Systems,2003,24(11):1950-1953(in Chinese)
赵军,王国胤,吴中福,等. 一种高效的属性核计算方法[J]. 小型微型计算机系统,2003,24(11):1950-1953

[18] Yang Ming, Sun Zhi-hui. Improvement of discernibility matrix and the computation of core[J]. Journal of Fudan University,2004,43(5):865-868(in Chinese)
杨明,孙志挥. 改进的差别矩阵及其求核方法[J]. 复旦学报(自然科学版),2004,43(5):865-868

[19] Liu Shao-hui, Sheng Qiu-jian, Shi Zhong-zhi. A New Method for Fast Computing Positive Region[J]. Journal of Computer Research and Development,2003,40(5):637-642(in Chinese)
刘少辉,盛秋戬,史忠植. 一种新的快速计算正区域的方法[J]. 计算机研究与发展,2003,40(5):637-642

[20] Xu Zhang-yan, Liu Zuo-peng, Yang Bing-ru, et al. A quick attribute reduction algorithm with complexity of max $\{O(|C||U|), O(|C|^2|U/C|)\}$ [J]. Chinese Journal of Computers,2006,29(3):391-399(in Chinese)
徐章艳,刘作鹏,杨炳儒,等. 一个复杂度为 max$\{O(|C||U|), O(|C|^2|U/C|)\}$的快速属性约简算法[J]. 计算机学报,2006,29(3):391-399

[21] Xu Zhang-yan, Yang Bing-ru, Song Wei. Quick Computing Core Algorithm Based on Discernibility Matrix[J]. Computer Engineer and Applications,2006,42(6):4-6(in Chinese)
徐章艳,杨炳儒,宋威. 一个基于差别矩阵的快速求核算法[J]. 计算机工程与应用,2006,42(6):4-6

[22] Xu Zhang-yan, Yang Bing-ru, Cai Wei-dong, et al. Quick algorithm for computing core based on the positive region [J]. Systems Engineering and Electronics,2006,28(12):1902-1905(in Chinese)
徐章艳,杨炳儒,蔡卫东,等. 一个基于正区域的快速求核算法[J]. 系统工程与电子技术,2006,28(12):1902-1905

[23] Wang Guo-yin. Rough set theory and knowledge acquisition [M]. Xi'an: Xi'an Jiao Tong University Press, 2011(in Chinese)
王国胤. Rough 集理论与知识获取[M]. 西安:西安交通大学出版社,2001