实验报告

无研153 2015310524 徐丰力 18515396742

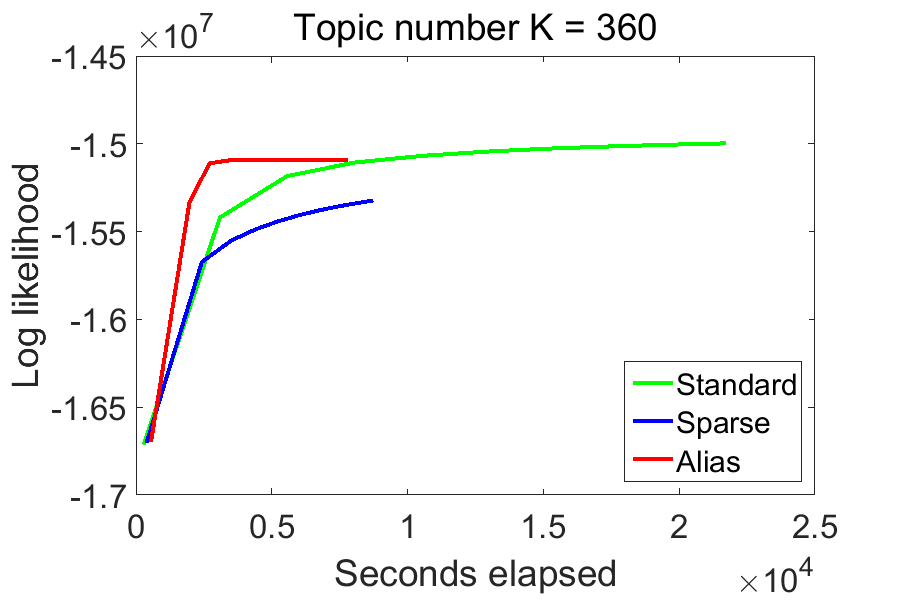
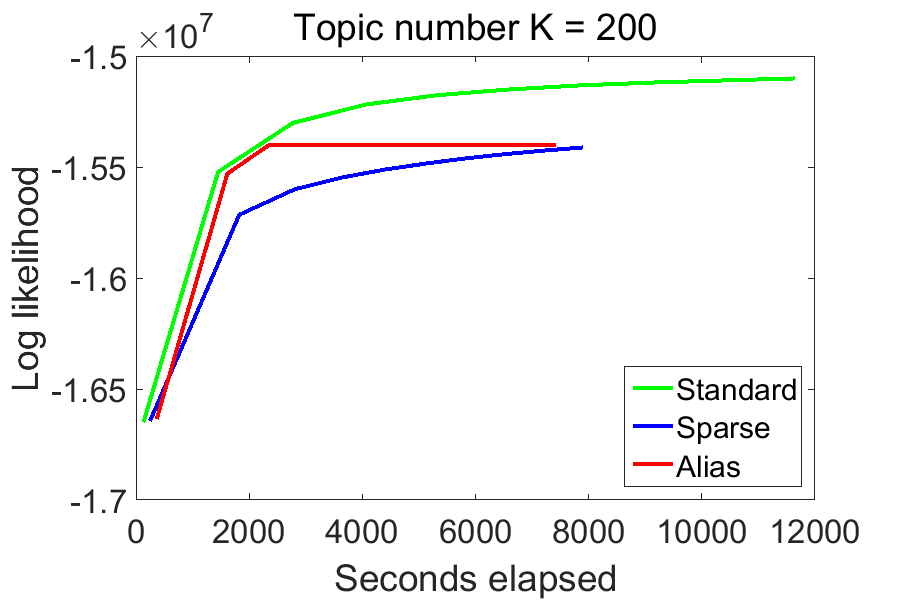
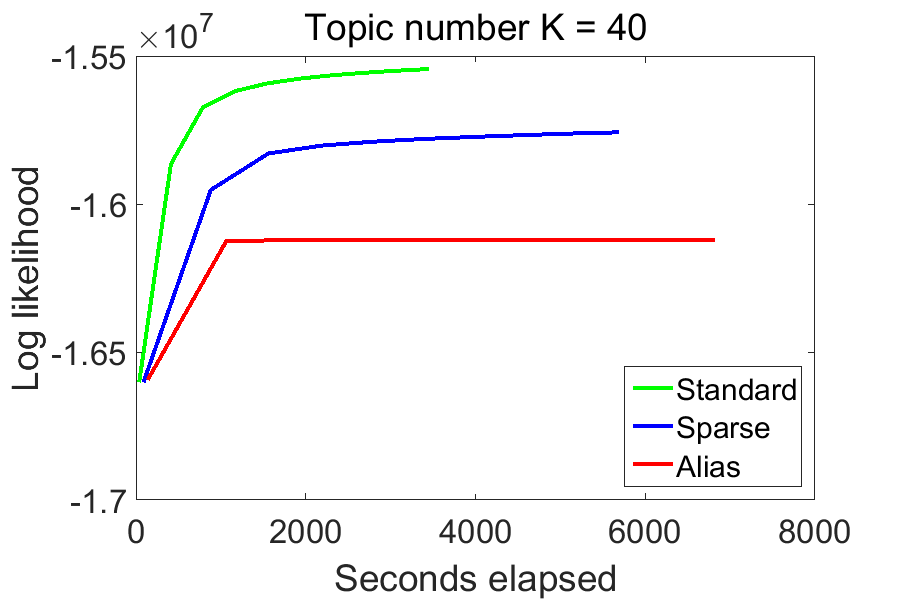
1. 实验原理

在本次实验中，我实现了LDA、SparseLDA和AliasLDA的算法，并在nips数据集上测试了算法的性能。下面分为数据预处理，LDA，SparseLDA和AliasLDA四个部分，分别简单介绍算法的原理以及具体的代码实现：

1. **数据预处理：**遍历指定目录下的所有文档，按文件名读入每篇文档。使用nltk库中的word\_tokenize 对文档进行分词，用PorterStemmer提取每个单词的词干。最后，根据停用词表过滤停用词，并用正则表达式去除非法单词（长度为1或者有非法符号）。
2. **LDA：**基于Gibbs采样的基础LDA。初始化时对所有单词随机分配主题，然后根据全条件概率公式迭代采样并更新每个单词的主题。每次采样的复杂度为O（K），整体复杂度为O（N\*I\*K）。其中，K为主题个数，N为所有单词个数（含重复），I为迭代次数。
3. **SparseLDA：**依旧是基于Gibbs采样的LDA模型。但利用了主题数很多时，文档和单词的主题分布很稀疏的特点，采用了稀疏存储结构，将每次采样的时间复杂度降低到了O（Kd+Kw）。整体时间复杂度为O（N\*I\*（Kd+Kw））。其中，Kd和Kw分别为文档和单词的主题个数（不为零的量）。在主题分布稀疏时，即Kd+Kw << K时，该算法在复杂度上有较大优势。
4. **AliasLDA：**采用了Alias Sampling的方法，将每次采样的时间复杂度降低到了O（Kd）。具体做法是将采样公式中与单词相关的部分构造Alias Table，用 Alias Sampling将这部分的平均采样复杂度降低为O（1）。 由于不是用权条件概率公式采样，这种训练方法退化为Metropolis-Hasting 采样法，但因为主题模型的概率分布变化很慢，因此采样时的拒绝率不会很高。整体复杂度为O（N\*I\*Kd）。在文档稀疏，单词不稀疏时较SparseLDA有明显优势。
5. 结果分析

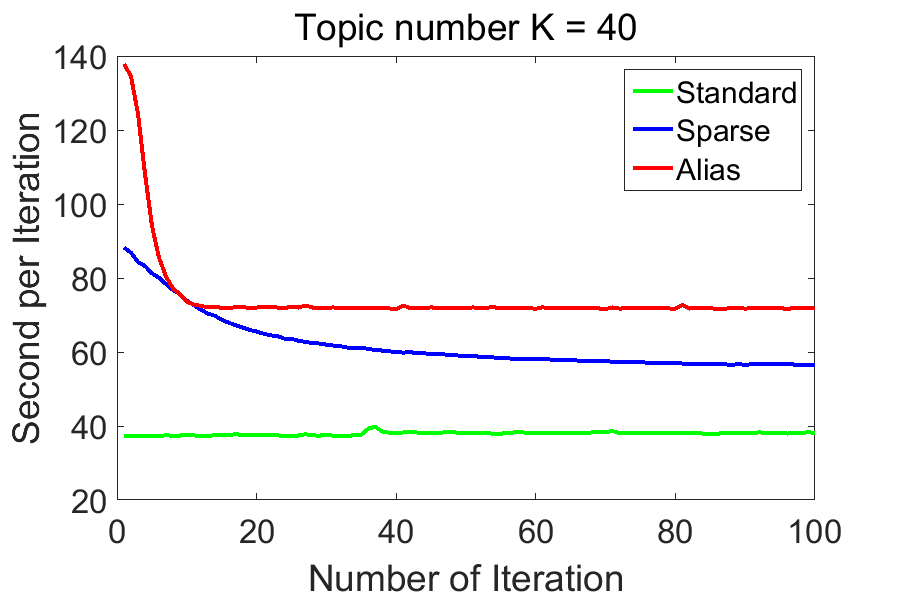
实现完代码后，我在nips数据集上对三种不同的算法进行了测试。收集了三种算法在不同主题数目下，迭代100次的性能数据：模型质量（Log-likelihood）和每次迭代消耗时间。在这一部分，我详细对比了不同算法间的性能差异，分为以下三个方面总结如下：

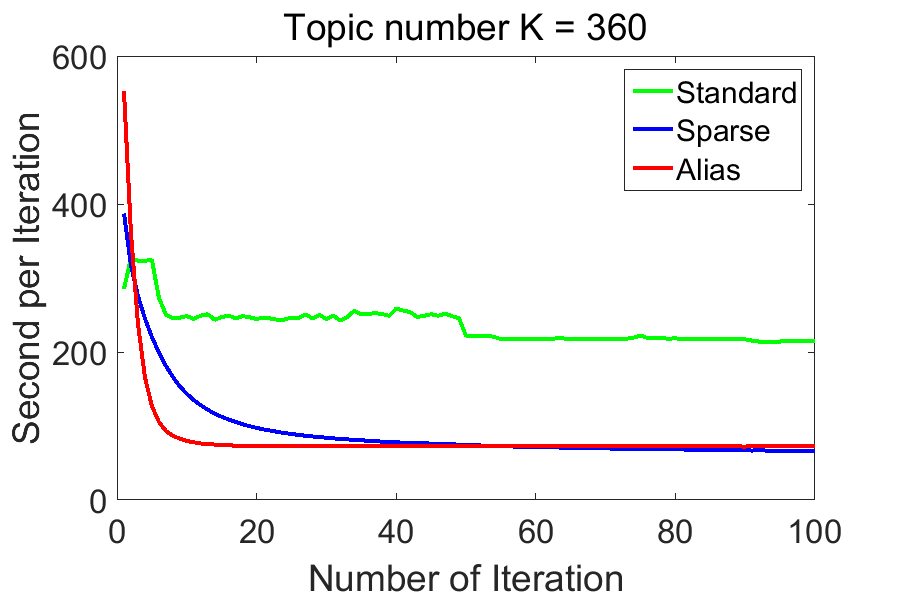
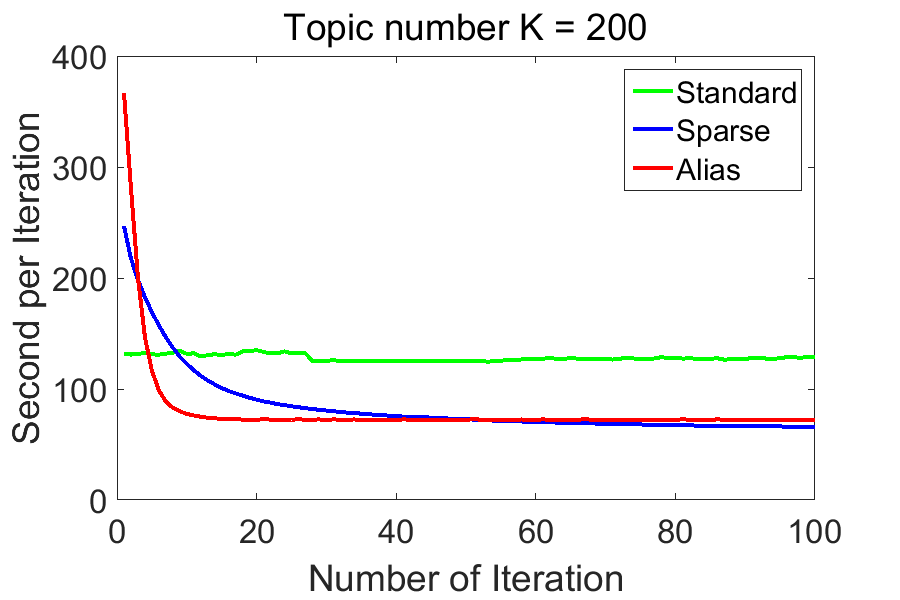
1. **模型质量 vs 累积耗时：**

****

以上三图为主题数分别为40, 200 和 360时三种算法的模型质量随累积消耗时间的变化曲线。可以看到，在主题数很小时（40），基本的LDA反而效率最高，这是由于此时文档和单词的主题分布均不稀疏，SparseLDA和AliasLDA算法的优势不明显，反而因为要维护复杂的数据结构有更多的时间开销。当主题数有40逐步变化为360时，SparseLDA和AliasLDA的性能开始逐步接近并超过基础的LDA，此时主题分布稀疏带来的优势开始变得明显。

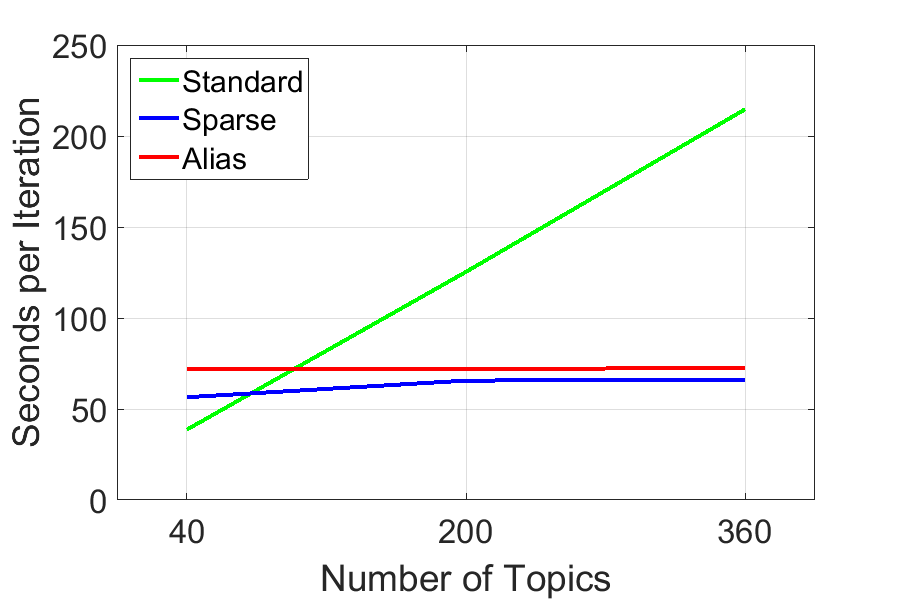
1. **迭代时间 vs 迭代次数：**

****

****

以上三图为主题数分别为40，200和360时，每次迭代耗时随迭代次数变化的曲线，可以看到普通LDA的迭代耗时基本不随迭代次数发生变化。而SparseLDA和AliasLDA的每次迭代耗时随着迭代次数的增加和显著下降。这是因为，这两种算法的迭代时间与文档和单词的主题分布的稀疏程度密切相关。而初始化时，由于随机分配主题导致文档和单词的主题分布均布稀疏，所以耗时很高，收敛后则耗时显著下降。

1. **收敛后迭代时间 vs 主题个数：**



上图为三种算法的收敛后迭代时间随主题数目变化的情况，可以看到普通LDA虽然在主题数为40时较另外两种有优势，但是随着主题数上升，迭代时间线性增长。而AliasLDA和SparseLDA的迭代时间基本上不随主题数的增加而增加，在主题数目多时有明显优势。这与理论分析的时间复杂度一致。

1. 总结

在这次实验中，我学习并实现了两种efficient LDA算法。实验结果发现，在主题数目较多时，SparseLDA和AliasLDA算法有较大优势， 在主题数目较少时，性能甚至反而不如普通的LDA模型。另外，我实现的SparseLDA和AliasLDA并没有取得论文中报告的性能增益，具体原因我认为主要有两点：一是主题主题数目还不够多，理论分析和实验结果明显可以看出主题数目越多时性能增益越明显，但由于单机计算能力有限，在本实验中最多只测试了360个主题，没有达到论文中1000个主题数左右的规模；二是数据集不够理想，Alias LDA重点针对的是文档分布很稀疏，单词不稀疏的数据，即有很多篇文档，每篇文档只有十几个单词。但这次我用的数据集，文档数只有1000多篇，每篇文档有几百到一千多个单词。因此，文档的主题分布并不特别稀疏，Alias LDA和Sparse LDA的性能增益并不明显。