基于 Topic Model 的中文文本分类

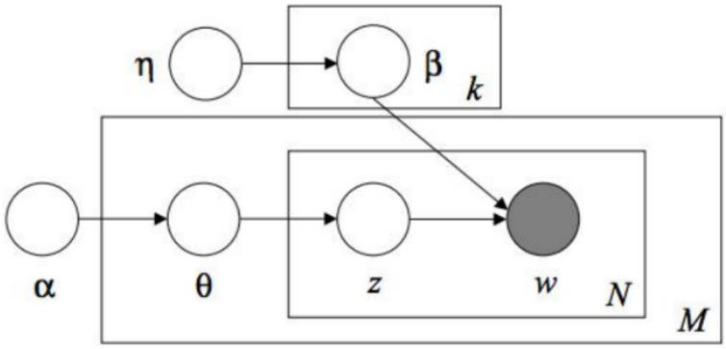
题目：从给定的语料库中均匀抽取 200 个段落（每个段落大于 500 个词），每个段落的标签就是对应段落所属的小说。利用 LDA 模型进行文本建模，并把每个段落表示为主题分布后进行分类。验证与分析分类结果。

1 LDA 模型

LDA(Latent Dirichlet Allocation)是一种文档主题生成模型，也称为一个三层贝叶斯概率模型，包含词、主题、和文档三层结构。所谓生成模型，我们认为一篇文章的每一个词都是通过“文章以一定的概率选择了某一主题，并从这个主题中以一定的概率选择某一词语”这个过程得到。文档到主题服从多项式分布，主题到词服从多项式分布。

对于语料库中的每篇文档，LDA 定义了如下的生成过程：

1. 对于每一篇文档，从主题分布中抽取一个主题。
2. 从上述被抽到的主题所对应的单词分布中抽取一个单词。
3. 重复上述过程直至遍历文档中的每一个单词。



1. Theta 文档-主题分布是多项式分布，该多项式分布的参数服从 Dirichlet 分布，该 Dirichlet 分布的参数为 alpha，theta 的每一行数据是一个 K 维向量（语料库共有 K 个主题），比如（1，0，0，1，0，1），表示该文档包含那些主题以及对应的概率。
2. Beta 主题-词语分布是多项式分布，该多项式分布的参数服从 Dirichlet 分布，该 Dirichlet 分布的参数为 nita；beta 的每一行是一个 V 维的向量，表示给主题包含哪些词语以及对应的概率。
3. Z 是从 theta 中抽取出了的一个主题，是一个 k 维向量，比如（0，0，0，1，0，0）.
4. W 是从 z 这个主题及其对应的词语中抽取出来的一个词语（观测值）
5. 重复 3-4 步骤直至遍历该文档中的每一个单词，然后遍历下一个文档直至 m 篇文档全部完成。
6. 选择 m 个待分类文档，再逐个抽完 n 个词后，还原观测值此时将图中最高的柱状图对

应的主题找出来分析这些主题中出现最高频的词语，根据这些词语，认为定义主题分类名称。

1.1 LDA 训练算法

1. 随机初始化 alpha、beta
2. 以下步骤迭代直至收敛：
   1) 对训练集中的每篇文档利用当前的 alpha 和 beta 值计算每篇文档的主题分布、每个词所属的主题分布。
   2) 积累所有文档中，属于主题 K 的词的个数，得到 gammas;以及词 i 属于主题 k 的次数，得到矩阵 batas；
   3) 根据当前的 gammas，利用 Newton-Raphson 迭代方法求得当前的最优 α 值；
   4) 对矩阵 betas 的列归一化，直接得到当前的 β 值，即每个主题的词的分布；
3. 输出达到收敛时的 α 和 β 的值

1.2 LDA 预测算法

1. 以平均分布初始化 K 维向量 nt，ntk 是当前文档中属于类别 k 的词的个数，nt 可视为未归一化的文档的主题分布；
2. 以下步骤迭代直到 nt 达到稳定：
   1) 根据当前的 α 值(决定主题的先验分布)，以及当前的 nt 值(当前文档的主题分布)，以及当前的 β 值(主题的词的分布)，计算文档中的各个词的主题分布，得到矩阵 q， qij=文档中的第 i 个词属于主题 k 的概率。
   2) 利用矩阵 q 的值更新向量 nt 的值。
3. 将 nt 归一化作为文档的主题分布，矩阵 q 则为文档中每个词的主题分布。

2 文本分类

文本分类是指在给定分类体系，根据文本内容自动确定文本类别的过程。最基础的分类是归到两个类别中，称为二分类问题，例如电影评论分类，只需要分为“好评”和“差评”。分到多个类别中的称为多分类问题，例如，把名字分类为法语名字、英语名字、西班牙语名字。

2.1 步骤

一般来说文本分类大致分为以下几个步骤：

1. 定义阶段：定义数据以及分类体系，具体分为哪些类别，需要哪些数据。
2. 数据预处理：对文档做分词、去停用词等准备工作。
3. 数据提取特征：对文档矩阵进行降维，提取训练集中最有用的特征。
4. 模型训练阶段：选择具体的分类模型以及算法，训练出文本分类器。
5. 评测阶段：在测试集上测试并评价分类器的性能。
6. 应用阶段：应用性能最高的分类模型对待分类文档进行分类。

3 实验过程

本次实验继续使用金庸先生的 16 本武侠小说作为数据集，利用 LDA 进行文本分类。

3.1 数据预处理

在这一环节中，删除文本的所有隐藏符号，删除所有的非中文字符，不考虑上下文关系的前提下删去所有的标点符号。以 jieba 库对中文语料进行分词。得到训练集。

训练集内容如下，共 59202 行：

```
多谢 大王 厚礼 命臣 奉 上 宝剑 一口 还 答此 剑 乃 敝国 新铸 谨供 大王 玩赏
庸人 自必 骂 他 糊涂 你们 又 怎能 明白 范先生 呢 便 亲自 前去 拜访 范 避而不见
宫门 走 去
青衣 剑士 取 的 纯 是 守势 招数 严密 竟 一招 也 不 还击 却令 三名 锦衫 剑士 无法 过
八名 身穿 青衣 的 汉子 手臂 挽 着 手臂 放喉 高歌 旁若无人 的 大踏步 过来
锦衫 剑士 突然 发足 疾奔 绕 着 青衣 剑士 的 溜溜 的 转动 脚下 越来越快 青衣
众 卫士 退 了 下去 范蠡 握 着 西施 的 手道 咱们 换上 庶民 的 衣衫 我 和 你 到 太
家里 还有 什么 人 阿青道 就是 我妈 和 我 两个 人 不 知道 我妈 肯不肯 来 我
尽 如此 剑 之利 但 观此 一端 足见 其余 最 令人 心忧 的 是 吴国 武士 群战 之术
为首 的 吴士 仰天大笑 说道 我们 从 姑苏 来到 会稽原 是 不想 再 活着 回去
```

3.2 运行 LDA 进行训练并验证

本次实验使用了 gensim 中的 corpora 和其中自带的 lda 模型进行训练。

```
    """构建词频矩阵，训练LDA模型"""
    dictionary = corpora.Dictionary(train)
    # corpus[0]: [(0, 1), (1, 1), (2, 1), (3, 1), (4, 1),...]
    # corpus是把每本小说ID化后的结果，每个元素是新闻中的每个词语，在字典中的ID和频率
    corpus = [dictionary.doc2bow(text) for text in train]
    #
    lda = models.LdaModel(corpus=corpus, id2word=dictionary, num_topics=16)
    # lsi = models.LsiModel(corpus=corpus,id2word=dictionary,num_topics=16)
    topic_list_lda = lda.print_topics(16)
    # topic_list_lsi = lsi.print_topics(16)
    print("以LDA为分类器的16个主题的单词分布为：\n")
    for topic in topic_list_lda:
        print(topic)
```

使用验证集进行验证，测试训练结果：

```
"""测试"""
file_test = "./cnews.test_jieba.txt"
news_test = open(file_test, 'r', encoding='UTF-8')
test = []
# 处理成正确的输入格式
for line in news_test:
   # line = line.split('\t')[1]
    #line = re.sub(r'[^\u4e00-\u9fa5]+', '', line)
    line = [word.strip() for word in line.split(' ')]
    test.append(line)

for text in test:
    corpus_test = dictionary.doc2bow((text))
    # print(corpus_test)
```

3.3 实验结果

16 个主题的词语分布结果如下：

```
(0, '0.038*"言语" + 0.024*"面前" + 0.024*"及" + 0.019*"觉" + 0.019*"或" + 0.015*"她们" + 0.013*"应道" + 0.012*"修为" + 0.011*"自幼" + 0.010*"服侍"')
(1, '0.067*"两位" + 0.020*"发作" + 0.018*"衣袖" + 0.015*"诀" + 0.013*"门户" + 0.013*"窜" + 0.013*"迅速" + 0.011*"无异" + 0.010*"偏" + 0.010*"大殿"')
(2, '0.084*"剑法" + 0.041*"无法" + 0.025*"未" + 0.024*"多半" + 0.018*"撞" + 0.016*"之色" + 0.010*"大门" + 0.010*"传授" + 0.010*"眼前" + 0.009*"倒退"')
(3, '0.068*"罢" + 0.050*"兄弟" + 0.025*"否则" + 0.020*"即" + 0.015*"显然" + 0.012*"别说" + 0.012*"按" + 0.011*"不如" + 0.010*"一道" + 0.010*"从此"')
(4, '0.042*"的" + 0.041*"了" + 0.039*"我" + 0.038*"道" + 0.032*"是" + 0.025*"你" + 0.024*"也" + 0.020*"说" + 0.019*"他" + 0.016*"那"')
(5, '0.023*"少女" + 0.018*"了" + 0.018*"那" + 0.015*"抢" + 0.013*"老人家" + 0.013*"正" + 0.013*"出" + 0.012*"一位" + 0.009*"在" + 0.009*"不及"')
(6, '0.022*"此刻" + 0.014*"在" + 0.014*"人物" + 0.014*"不得" + 0.014*"其中" + 0.013*"如此" + 0.012*"的" + 0.012*"是" + 0.012*"出" + 0.010*"抓"')
(7, '0.032*"写" + 0.024*"皇帝" + 0.021*"一日" + 0.018*"相见" + 0.017*"怪" + 0.016*"右掌" + 0.016*"位" + 0.015*"那姓" + 0.015*"旁观" + 0.014*"般的"')
(8, '0.017*"出" + 0.014*"在" + 0.013*"武林中" + 0.012*"的" + 0.009*"便" + 0.008*"下" + 0.008*"得" + 0.008*"往" + 0.008*"与" + 0.008*"若"')
(9, '0.060*"的" + 0.042*"了" + 0.032*"他" + 0.025*"是" + 0.023*"在" + 0.014*"这" + 0.013*"她" + 0.011*"又" + 0.011*"便" + 0.010*"将"')
(10, '0.030*"此处" + 0.026*"房中" + 0.023*"惊" + 0.021*"伤势" + 0.020*"生气" + 0.017*"转念" + 0.016*"见识" + 0.014*"一颗" + 0.012*"不管" + 0.011*"连声"')
(11, '0.155*"你" + 0.096*"我" + 0.080*"道" + 0.028*"了" + 0.016*"叫" + 0.012*"好" + 0.011*"不" + 0.011*"说道" + 0.010*"他" + 0.010*"笑"')
(12, '0.033*"不料" + 0.017*"决不能" + 0.016*"落入" + 0.015*"猛" + 0.012*"发" + 0.011*"有意" + 0.011*"那边" + 0.011*"一层" + 0.010*"一年" + 0.009*"臂"')
(13, '0.025*"得" + 0.025*"左手" + 0.020*"长剑" + 0.018*"一声" + 0.018*"右手" + 0.015*"已" + 0.015*"向" + 0.015*"听" + 0.011*"之声" + 0.010*"只"')
(14, '0.060*"师哥" + 0.043*"武林" + 0.039*"张" + 0.026*"怎能" + 0.019*"纷纷" + 0.019*"而已" + 0.016*"西域" + 0.014*"劈" + 0.012*"四下" + 0.011*"通红"')
(15, '0.029*"尽数" + 0.026*"也好" + 0.018*"老子" + 0.017*"断" + 0.016*"抵挡" + 0.015*"足" + 0.014*"剩下" + 0.012*"呸" + 0.012*"是从" + 0.012*"一副"')
```

主题	词及其对应的概率									
0	0.038 言语	0.024面前	0.024 及	0.019 觉	0.019 或	0.015她们	0.013应道	0.012修为	0.011 自幼	0.010服饰
1	0.067 两位	0.020发作	0.018衣袖	0.015 诀	0.013门户	0.013 窜	0.013迅速	0.011 无异	0.010 偏	0.010大殿
2	0.084 剑法	0.041无法	0.025 未	0.024多半	0.018 撞	0.016之色	0.010大门	0.010传授	0.010眼前	0.009倒退
3	0.068 罢	0.050兄弟	0.025否则	0.020 即	0.015显然	0.012别说	0.012 按	0.011 不如	0.010一道	0.010从此
4	0.042 的	0.041 了	0.039 我	0.038 道	0.032 是	0.025 你	0.024 也	0.020 说	0.019 他	0.016 那
5	0.023 少女	0.018 了	0.018 那	0.015 抢	0.013老人家	0.013 正	0.013 出	0.012一位	0.009 在	0.009不及
6	0.022 此刻	0.014 在	0.014人物	0.014不得	0.014其中	0.013如此	0.012 的	0.012 是	0.012 出	0.010 抓
7	0.032 写	0.024皇帝	0.021一日	0.018相见	0.017 怪	0.016右掌	0.016 位	0.015那姓	0.015旁观	0.014般的
8	0.017 出	0.014 在	0.013武林中	0.012 的	0.009 便	0.008 下	0.008 得	0.008 往	0.008 与	0.008 若
9	0.060 的	0.042 了	0.032 他	0.025 是	0.023 在	0.014 这	0.013 她	0.011 又	0.011 便	0.010 将
10	0.030*" 此处	0.026*" 房中	0.023*"惊	0.021*" 伤势	0.020*" 生气	0.017*" 转念	0.016*" 见识	0.014*" 一颗	0.012*" 不管	0.011*" 连声
11	0.155 你	0.096 我	0.080 道	0.028 了	0.016 叫	0.012 好	0.011 不	0.011 说道	0.010 他	0.010 笑
12	0.033 不料	0.017决不能	0.016落入	0.015 猛	0.012 发	0.011 有意	0.011 那边	0.011 一层	0.010一年	0.009 臂
13	0.025 得	0.025左手	0.020长剑	0.018一声	0.018右手	0.015 已	0.015 向	0.015 听	0.011之声	0.010 只
14	0.060 师哥	0.043武林	0.039 张	0.026怎能	0.019纷纷	0.019而已	0.016西域	0.014 劈	0.012四下	0.011通红
15	0.029 尽数	0.026也好	0.018老子	0.017 断	0.016抵挡	0.015 足	0.014剩下	0.012 呸	0.012是从	0.012一副

3.4 测试结果

认为选取几本小说内的一些段落，将其文本进行预处理后作为测试集，测试 LDA 模型对于文本的分类效果。得到不同测试段落的主题分布。

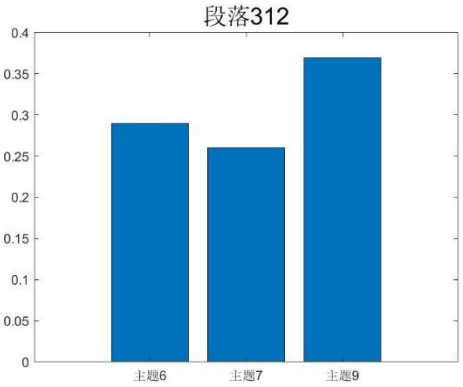
这里随机选取几段，将其分词后改成一行，结果如下：（这里仅展示了一段中的部分片段）

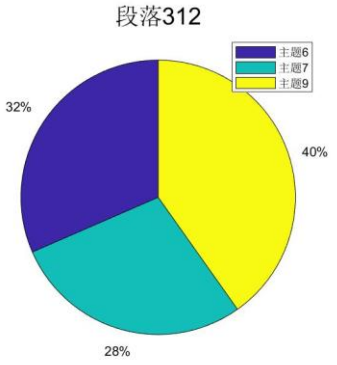
```
两人 一搭上手 顷刻间 拆了 三十来招，青衣剑士 被他 沉重的
张无忌 轻轻 推开 房门 揭开 门帘 但见 房内 黑沉沉
徐天宏 见 他 着力 办事 十分 义气 不住 道谢 上官毅山
```

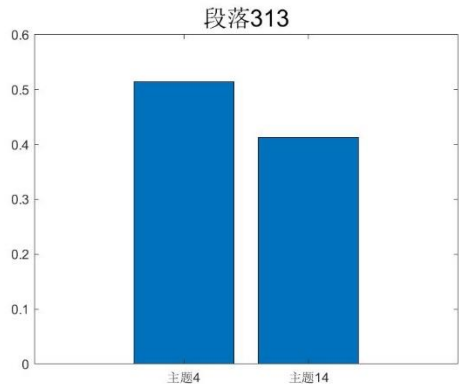
```
[(3, 0.087448455), (11, 0.68150634), (13, 0.16854005)]

312
的主题分布为:

[(6, 0.28970832), (7, 0.25973582), (9, 0.36928648)]

313
的主题分布为:

[(4, 0.51415384), (14, 0.4129165)]

314
的主题分布为:

[(6, 0.9278775)]

315
的主题分布为:

[(2, 0.16583113), (4, 0.06562563), (11, 0.122836635), (12, 0.053150058), (13, 0.054107867), (14, 0.5071738)]

316
的主题分布为:

[(0, 0.031253252), (1, 0.031253252), (2, 0.5312012), (3, 0.031253252), (4, 0.031253252), (5, 0.031253252), (6, 0.0312532
8), (7, 0.031253252), (8, 0.031253252), (9, 0.031253252), (10, 0.031253252), (11, 0.031253252), (12, 0.031253252), (13,
0.031253252), (14, 0.031253252), (15, 0.031253252)]
```

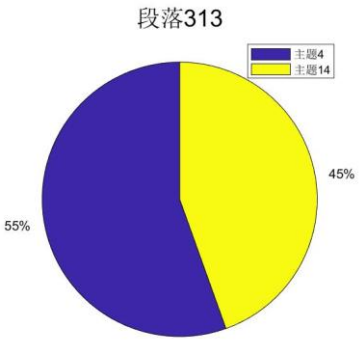
本次实验共选取了 317 个段落进行测试，测试结果如上图所示：（这里选取最后 5 个进行分析）。

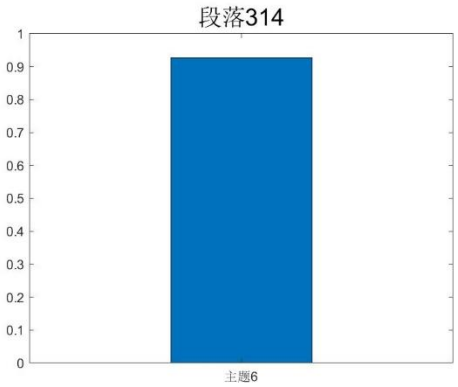
为了直观表示，将以上主题分布的预测结果数据导入 MATLAB 进行绘图分析。

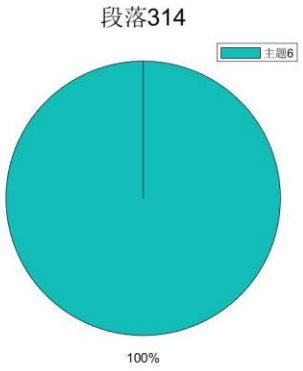


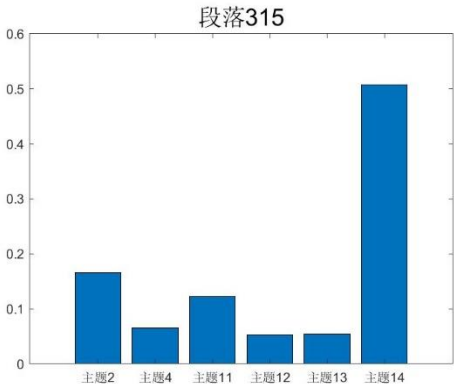


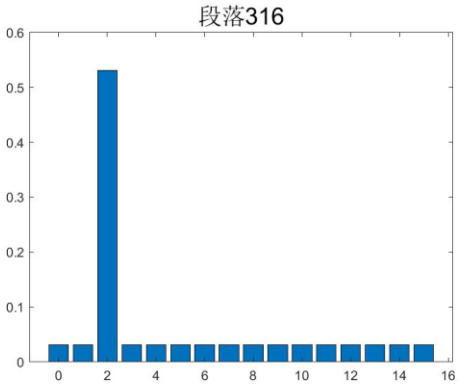


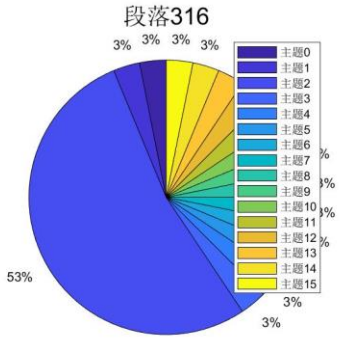












3.5 结果分析

从段落 316 的结果我们可以看出，该段落由主题 2 生成的概率最大，占到了总概率的百分之 50 以上。而段落 314 几乎全部由主题 6 所生成；段落 312，和段落 313 由其中几种主题所生成的概率比较平均。

4 参考文献

https://blog.csdn.net/weixin_42663984/article/details/116264233