深度学习与自然语言处理报告

19376252 张一辉
tbgsection@qq.com

摘要

本文是深度学习与自然语言处理的第三次实验报告，实验内容为基于 LDA（Latent Dirichlet Allocation）模型，使用给定的语料库在 16 本小说中均匀地抽取 200 个段落，段落标签为段落所属的小说，对文本进行预处理，建模和评估。并且选择不同的主题数量和不同基本单元进行建模比对分析。

引言

LDA 是一种文本分析算法，全称为 Latent Dirichlet Allocation，即潜在狄利克雷分配。它可以自动地将一篇文档中的词语划分到不同的主题中，每个主题由一组词语组成。LDA 算法可以用于文本分类、信息检索、情感分析等自然语言处理任务。本次实验以 16 本小说中抽取的段落作为建模使用的数据集，进行分词，去停词等预处理步骤。根据情况选定主题数进行建模，使用 Gibbs 算法进行模型的求解。对于模型的验证，使用 Kmeans 聚类方法对建模结果进行分类，对比段落归属的小说进行准确度的验证，评估模型。

研究方法

M1: LDA 模型

隐含狄利克雷分布（Latent Dirichlet allocation，LDA），是一种由 David Blei, Andrew Ng and Michael I. Jordan 提出的用于文本分析的生成概率模型。它可以将文档看作是多个主题的混合，并将每个主题看作是一组概率分布，其中每个单词都有特定的概率与该主题相关联。在训练过程中，LDA 会对每个文档分配主题，并对每个单词分配一个主题。

LDA 模型的主要思想是：假设文档是由若干个主题组成的，每个主题包含一组单词，并且每个单词都有一定的概率与该主题相关联。文档中的每个单词都是从某个主题中随机生成的。假设我们有一个由 N 个文档组成的文本集合，LDA 模型的目标就是通过对这些文档的分析来学习主题的分布。LDA 模型的输入是一个文档集合，输出是每篇文档的主题分布和每个主题的单词分布。在 LDA 模型中，每个单词都有一个隐含的主题，每个主题又有一个概率分布，从而形成了一个文档中所有单词的主题分布。LDA 模型的训练过程是通过迭代的方式来优化主题分布和单词分布的参数，使得模型能够最好地解释文档集合中的单词分布。

LDA 模型在文本挖掘、信息检索、社交网络分析等领域都有广泛的应用。它可以用于文本分类、主题分析、情感分析等任务，也可以用于推荐系统、广告推荐等领域。LDA 模型具有很好的可解释性和灵活性，可以根据具体的应用场景进行调整和扩展。

M2: Gibbs 采样

Gibbs 采样是求解 LDA 模型的经典算法之一，它通过对隐变量的采样来估计模型的参数。具体来说，Gibbs 采样通过反复迭代对文档中的每个单词进行采样，从而获得模型的参数估计。

在 LDA 模型中，Gibbs 采样的过程如下：

1. 初始化每个单词的主题，可以随机选择一个主题或者根据单词在语料库中的出现频率进行分配；
2. 对于每一个文档中的每一个单词，固定其它单词的主题，然后根据条件概率分布采样一个新的主题。具体来说，计算该单词在不同主题下的概率，然后根据概率进行采样；
3. 重复步骤 2，直到所有单词的主题都被更新过一遍；
4. 重复步骤 2 和步骤 3，直到模型的参数估计收敛或达到指定的迭代次数。

在进行 Gibbs 采样时，需要计算每个单词在不同主题下的概率，对于一个单词w，它在主题k下的概率可以表示为：

$$p(z_i = k|w_i = w, z_{-i}, w, \alpha, \beta) = \frac{n_{wk,-i} + \beta}{n_{k,-i} + W\beta} \times \frac{n_{dk,-i} + \alpha}{n_{d,-i} + K\alpha}$$

其中，$n_{wk,-i}$表示主题k中单词w的计数，但不包括当前单词w_i；$n_{k,-i}$表示主题k中单词的总数，但不包括当前单词w_i；$n_{dk,-i}$表示文档d中属于主题k的单词计数，但不包括当前单词w_i；$n_{d,-i}$表示文档d中所有单词的计数，但不包括当前单词w_i；W表示词汇表中的单词总数；K表示主题的个数；α和β分别表示文档-主题分布和主题-单词分布的超参数。

在每次迭代中，Gibbs 采样会根据上述公式计算每个单词在不同主题下的概率，并根据概率进行采样，更新每个单词的主题。经过多次迭代后，Gibbs 采样会得到 LDA 模型的参数估计。

M3: Kmean 聚类

K 均值聚类（K-means clustering）是一种常见的无监督学习算法，用于将具有相似特征的数据点分组到不同的类中。该算法的目标是最小化数据点到它们所属类中心的距离平方和，同时最大化不同簇之间的距离。

实验内容

数据处理

根据实验要求对由 16 本小说组成的文本均匀抽取 200 个段落，每个段落标签为文本所属小说。按照要求对语料库中每本小说抽取 13 个段落，这样就组成 208 个段落作为数据。将段落的序号除以 13 便得到了段落的标签，对应的顺序为 inf.txt 中小说名称的顺序。这样便得到了原始数据。

将原始数据进行预处理，首先在文本中删除来自小说网站的广告内容，这样使得结果更加精确，随后根据要求进行分词或者分字，分词使用 jieba 库进行分词。对于分词完成后的结果，将出现在 cn_stopwords.txt 停词表中的停词移除，最后再移除标点符号换行符等非中文文本后便完成了数据的预处理。

Gibbs 采样

使用 Gibbs 采样算法求解 LDA 模型，设定主题数为 20，迭代次数为 35 次
得到结果如下，以下为每个 topic 词频降序排列前 5 的词

topic0	剑士	116	剑	88	便	85	长剑	73	范蠡	67
topic1	麽	261	著	151	李文秀	120	中	109	听	100
topic2	便	66	想	55	众人	50	听	42	韦小宝	41
topic3	中	61	皇帝	45	做	41	一个	36	天下	35
topic4	武功	85	中	68	便	65	见	59	张召重	46
topic5	师父	85	便	79	令狐冲	75	中	71	范蠡	71
topic6	两人	127	胡斐	96	中	88	杨过	72	听	59
topic7	袁承志	134	便	113	笑	93	一声	93	见	76
topic8	陈家洛	99	石破天	53	事	51	会	48	中	45
topic9	便	128	韦小宝	90	杀	73	事	69	见	59

208 个段落建模得到的 topic 分布为

[[0.01402806 0.34468938 0.29859719 ... 0.10220441 0.03206413 0.0501002]
 [0.01603206 0.85971944 0.01202405 ... 0.04008016 0.00601202 0.01202405]
 [0.00601202 0.40881764 0.00801603 ... 0.44689379 0.00801603 0.04408818]
 ...
 [0.11222445 0.01002004 0.01402806 ... 0.02805611 0.00200401 0.01002004]
 [0.15631263 0.0260521 0.00801603 ... 0.01002004 0.00801603 0.01402806]
 [0.06206897 0.02068966 0.0183908 ... 0.0137931 0.01609195 0.0183908]]

Kmeans 聚类

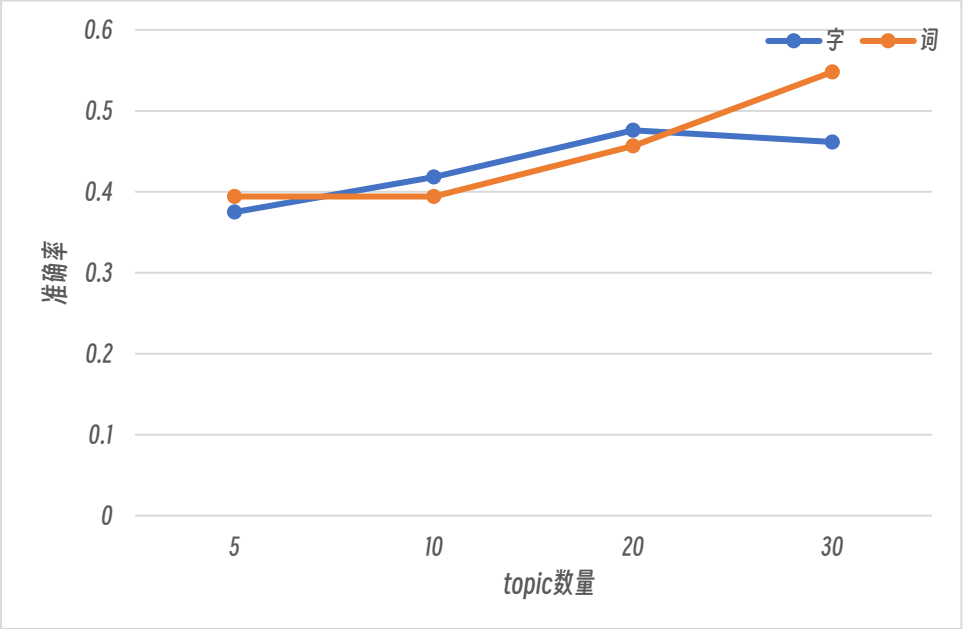
为了对 LDA 建模进行评估，对以上 topic 的分布向量进行聚类，观察来自同一部小说的段落是否能被分类至同一类

	标签												
小说 1	13	1	11	13	1	1	1	1	13	1	1	1	1
小说 2	5	11	11	12	4	11	11	11	11	2	11	9	14
小说 3	9	8	11	5	8	9	8	8	8	8	7	5	12
小说 4	11	4	9	9	15	11	5	5	5	15	4	2	4
小说 5	14	6	11	6	11	11	0	12	5	5	1	5	11
小说 6	14	14	1	10	3	9	14	14	0	4	6	0	14
小说 7	10	4	4	12	6	4	5	0	4	11	2	0	4
小说 8	14	0	13	13	12	11	9	13	10	8	5	4	10
小说 9	7	9	5	12	9	5	9	12	5	12	2	7	7
小说 10	4	8	0	14	11	7	2	5	7	6	7	6	0
小说 11	5	15	12	10	4	1	4	15	7	11	15	15	0
小说 12	8	5	5	4	4	7	10	12	5	4	4	10	12
小说 13	13	13	8	12	1	8	1	1	1	1	13	8	13
小说 14	14	0	6	6	6	8	8	8	6	6	3	10	12
小说 15	6	6	1	2	6	6	6	6	11	2	6	2	6
小说 16	3	3	3	3	3	3	3	3	10	4	4	4	4

可以看到，整体来说，在一些小说中，段落能被分为同一个标签，但在有的小说中，结果并不理想，这可能与我们选取的 topic 数量和迭代次数有关

选取不同参数下的结果

选取 topic 数量为 5，10，20，30，并且进行分词和分字的 LDA 建模，将聚类后的结果以出现次数多的标签作为正确值进行指标量化计算。



结论

针对不同 topic 数量，LDA 模型的性能有所差别，当分类数量过少时，比较难区分不同小说，此时正确率较低，当 topic 数量提升后，能够更加准确地区分不同小说。此外，当 topic 数量过多时，分类的结果可能会产生不稳定，可能将不同小说归为同类，因此，需要根据实际情况选择 topic 的数量

在实验过程中，我们分别选择了“词”和“字”作为基本单元进行分类。以“字”分类时在 topic 不高的情况下，能够实现较为不错的分类效果，而以“词”作为基本单元时，在 topic 较高时能够有着不错的分类效果。实际情况中，需要结合 topic 数量的选择，以及停词表准确性来选择基本单元。