Report of Deep Learning for Natural Language Processing Ⅱ

Yucheng Wang

wang_eil@126.com

Abstract

本文从金庸的 16 篇小说库中均匀抽取了 1000 个段落作为数据集（每个段落可以有 K 个 token, K 可以取 20,100,500, 1000, 3000）。利用 LDA 模型在给定的语料库上进行文本建模，主题数量为 T（T 可以取 10,20,50,100），把每个段落表示为主题分布后使用随机森林分类器进行分类。

比较了在设定不同的主题个数 T 的情况下，分类性能的差异；以"词"和以"字"为基本单元下分类性能的差异；不同的取值的 K 的短文本和长文本，分类性能的差异，以及选择不同大小的语料库时分类性能的差异。

Introduction

LDA，即 Latent Dirichlet Allocation（潜在狄利克雷分配），是一种常用的文本主题模型，用于发现文档集合中的潜在主题以及每个文档所包含的主题分布。在自然语言处理领域有广泛的应用，特别是在文本挖掘、信息检索和推荐系统等任务中。

LDA 假设每个文档都由多个主题组成，并且每个主题都由一组单词组成。模型通过以下过程工作：

步骤一—初始化：随机地为每个文档中的每个单词分配一个主题，以及为每个主题中的每个单词分配一个概率。

步骤二—迭代：迭代地更新每个单词的主题分配，以及每个主题中的单词概率，直到达到收敛条件。

步骤三—输出：得到每个文档的主题分布以及每个主题的单词分布。

通过 LDA 模型对段落生成的主题分布，与每个段落对应的标签输入随机森林分类器进行分类，可以实现对 LDA 生成效果的测试。

Methodology

Part1：文档抽取

由于每篇小说的字数不同，从所有小说中抽取 1000 个段落需要按照比例进行分配。首先读取每本小说的字数，然后按比例分配从每本小说抽取的段落数量。最后统计分配后的总数，如果不足 1000 段（可能因为小数舍弃的问题），从《鹿鼎记》中抽取缺少的段落补足 1000 段。

抽取段落前，首先对文档进行预处理，在预处理中去除标点符号、停用词、隐藏符号和非中文字符。对预处理后的文档，进行分词，按照字和词分别抽取 1000 段到 csv 文件中存储，并将对应文件名作为段落的标签，每个段落内使用逗号分隔字与字、词与词。

Part2：LDA 模型构建

LDA（Latent Dirichlet Allocation）是一种用于发现文本集合中潜在主题的统计模型。它的构建步骤包括：

数据准备：收集文本数据集，并进行预处理，该步骤已在 Part1 完成。

文档表示：将文档表示为词频或 TF-IDF 向量，其中向量的长度等于词汇表中的词汇数量，而每个维度对应于词汇表中的一个词汇，在文档中的出现频率或 TF-IDF 值。

初始化模型参数：设置主题数量，并随机初始化文档的主题分布和主题的单词分布。

迭代优化：使用 Gibbs 采样等方法迭代更新模型参数，直到收敛。

LDA 模型为理解和分析大规模文本数据提供了有力工具，可应用于各种自然语言处理任务。

Part3：使用随机森林进行分类

随机森林（Random Forest）是一种集成学习方法，用于解决分类和回归问题。它通过构建多个决策树，并将它们的输出结合起来进行预测，从而提高了模型的性能和鲁棒性。随机森林的基本组成部分是决策树。决策树是一种基于树结构的分类模型，通过对数据进行递归分割，将数据划分为不同的类别。通过构建多个

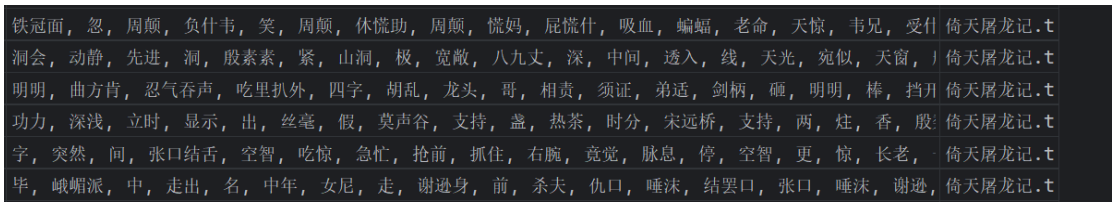
决策树，并将它们的预测结果进行组合，来提高整体模型的性能。本文中使用 90%的段落作为训练集，10%作为测试集，并进行 10 次交叉验证。

Experimental Studies

经过统计每篇小说的字数，在使用 16 篇小说作为源时，每篇小说抽取的段落数目如下：

名称	段落数目	名称	段落数目
三十三剑客图	7	书剑恩仇录	61
侠客行	41	倚天屠龙记	113
天龙八部	139	射雕英雄传	106
碧血剑	57	白马啸西风	7
神雕侠侣	114	笑傲江湖	111
越女剑	2	连城诀	26
飞狐外传	51	雪山飞狐	15
鸳鸯刀	4	鹿鼎记	146

进行段落抽取后的 csv 文件如图所示：



以词为单位：K=20,100,500,1000,2000；T=10,20,50,100 进行实验，分类准确度如下：

K \ T	10	20	50	100
20	0.084	0.152	0.110	0.150
100	0.162	0.148	0.164	0.167
500	0.200	0.220	0.304	0.314
1000	0.218	0.288	0.364	0.364
2000	0.332	0.457	0.628	0.606

以字为单位：K=20,100,500,1000,2000；T=10,20,50,100 进行实验，分类准确度如下：

K \ T	10	20	50	100
20	0.126	0.164	0.128	0.180
100	0.162	0.178	0.126	0.136
500	0.336	0.418	0.444	0.576
1000	0.342	0.483	0.599	0.681
2000	0.442	0.521	0.746	0.713

尝试减少文件数量到 6 个（碧血剑、飞狐外传、鹿鼎记、天龙八部、笑傲江湖、倚天屠龙记），抽取 1000 个段落，每个段落 2000 个词，主题数量 T =10,20,50,100，分类准确度结果如下：

K \ T	10	20	50	100
2000	0.374	0.53	0.558	0.588

每个段落 2000 个字的分类准确度如下：

K \ T	10	20	50	100
2000	0.564	0.724	0.78	0.83

Conclusion

通过分析实验得到的结果可以总结出如下结论：

① 以字为单位的 LDA 模型相比以词为单位的 LDA 模型，在随机森林分类器中的分类效果更好；

② Token 数量越多，分类效果越好；

③ LDA 模型主题越多，分类效果越好；

④ 如果选择更少的文件数目作为源，在以字为单位的 LDA 模型中，随机森林分类器的分类效果显著提升，在以词为单位的模型中没有显著提升。

通过实验，验证了 LDA 模型在短文本上的效果不好。