# Report of Deep Learning for Natural Langauge Processing

# Homework_2

Pan Yao
1239388514@qq.com

## Abstract

从给定的中文语料库中均匀抽取 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），每个段落的标签就是对应段落所属的小说。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，并把每个段落表示为主题分布后进行分类（分类器自由选择），分类结果使用 10 次交叉验证（i.e. 900 做训练，剩余 100 做测试循环十次）。分别针对在设定不同的主题个数 T 的情况，以“词”和以“字”为基本单元的情况下，查看分类性能的变化与结果的差异，同时根据不同的取值的 K 的短文本和长文本，比较主题模型性能的差异。

## Introduction

LDA 模型

LDA（Latent Dirichlet Allocation）是一种流行的主题模型，常用于自然语言处理和文本挖掘领域，特别是在处理文档集合或语料库时用于发现文档内潜在的主题结构。这个模型是由 Blei, Ng, and Jordan 在 2003 年提出的。

基本原理

LDA 是一种生成模型，它假设文档是由多个主题按一定比例混合而成的，而每个主题又是由多个词构成的。模型的目标是根据文档集合推断出潜在的主题结构以及每个文档中主题的分布。

模型假设

每个文档表示为一个词袋：文档中的词序信息被忽略，只考虑词的频率。文档中的每个词都是通过随机过程生成的：

首先，选择一个主题，这一选择依赖于文档特定的主题分布。

然后，从这个主题对应的词分布中选择一个词。

参数

主题分布：文档中各个主题出现的概率分布，服从 Dirichlet 分布。

词分布：每个主题中各个词出现的概率分布，也服从 Dirichlet 分布。

应用

LDA 广泛应用于文档分类、信息检索、情感分析和文档摘要等领域。通过分析文档生成的主题，可以对大规模文本数据进行有效的组织、概括和索引。

# Experiment

针对主题个数 T 和 token 个数 K 分别设置为[10, 25, 50, 100, 200, 500, 1000]和[20, 100, 500, 1000, 3000]，并使用随机森林作为分类器，查看以字为基本单元和以词为基本单元的情况下分类性能的变换。

# Experimental Studies

当设定 K=100 时，分别以词和字作为基本单元，使用随机森林分类器的分类性能变化如表 1 所示：

| Unit | K | T | Train Accuracy | Test Accuracy |
|---|---|---|---|---|
| word | 100 | 50 | 0.764333333 | 0.107 |
| word | 100 | 100 | 0.884333333 | 0.12 |
| word | 100 | 200 | 0.928777778 | 0.115 |
| word | 100 | 500 | 0.984555556 | 0.111 |
| word | 100 | 1000 | 0.998222222 | 0.149 |
| char | 100 | 50 | 0.998222222 | 0.456 |
| char | 100 | 100 | 0.998222222 | 0.458 |
| char | 100 | 200 | 0.997222222 | 0.451 |
| char | 100 | 500 | 0.996444444 | 0.49 |
| char | 100 | 1000 | 0.987 | 0.516 |

表 1 以词和字作为基本单元，不同主题数下分类性能

当设定 T=100 时，分别以词和字作为基本单元，使用随机森林分类器的分类性能变化如表 1 所示：

| Unit | K | T | Train Accuracy | Test Accuracy |
|---|---|---|---|---|
| word | 20 | 200 | 0.829333333 | 0.132 |
| word | 100 | 200 | 0.928777778 | 0.115 |
| word | 500 | 200 | 0.991555556 | 0.372 |
| word | 1000 | 200 | 0.997 | 0.672 |
| word | 3000 | 200 | 1 | 0.893 |
| char | 20 | 200 | 0.907444444 | 0.152 |
| char | 100 | 200 | 0.997222222 | 0.451 |
| char | 500 | 200 | 1 | 0.794 |
| char | 1000 | 200 | 1 | 0.844 |
| char | 3000 | 200 | 1 | 0.919 |

表 2 以词和字作为基本单元，不同 token 下分类性能

由表 1 可知，以随机森林作为分类器，当 token 固定时，topic 数越多，准确率会更高。
由表 2 可知，以随机森林作为分类器，当 topic 固定时，token 数越多，准确率会更高。
由表 1 和表 2 可知，以字为基本单元的分类准确率比以词为基本单元的分类准确率更高。