**Report of Deep Learning for Natural Langauge Processing**

Yuxiang Cheng ZY2303301

cyx1313@126.com

**Abstract**

给定的语料库中均匀抽取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（5, 10,25, 50, 100），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）。首先在设定不同的主题个数T和K的情况下，观察分类性能变化，然后以word和char为基本单元下比较分类结果的差异，并用了SVM、XGB、随机森林分类模型进行横行比较。

**Introduction**

LDA模型

LDA是自然语言处理中非常常用的一个主题模型，全称是隐含狄利克雷分布（Latent Dirichlet Allocation），简称LDA。作用是将文档集中每篇文档的主题以概率分布的形式给出，然后通过分析分到同一主题下的文档抽取其实际的主题（模型运行结果就是一个索引编号，通过分析，将这种编号赋予实际的意义，通常的分析方法就是通过分析每个topic下最重要的term来进行总结归纳），根据主题分布进行主题聚类或文本分类。

分类模型

XGBoost（eXtreme Gradient Boosting）是一种高效且强大的机器学习算法，用于解决分类和回归问题。它是基于梯度提升框架的集成学习方法，通过组合多个弱学习器（通常是决策树）来构建一个更强大的模型。

支持向量机（support vector machines, SVM）是一种二分类模型，它的基本模型是定义在特征空间上的间隔最大的线性分类器，间隔最大使它有别于感知机；SVM还包括核技巧，这使它成为实质上的非线性分类器。SVM的的学习策略就是间隔最大化，可形式化为一个求解凸二次规划的问题，也等价于正则化的合页损失函数的最小化问题。SVM的的学习算法就是求解凸二次规划的最优化算法。

随机森林（Random Forest）是一种集成学习算法，用于解决分类和回归问题。它由多个决策树组成，每个决策树独立地进行预测，并通过投票或平均等方式来确定最终的预测结果。

**Methodology& Experimental Studies**

Experiment One:

设定 k 为 3000 ，并使用各个分类器的分类性能的变化如下表所示

|  |  |  |  |
| --- | --- | --- | --- |
| 主题 | 模型 | 训练精度 | 测试精度 |
| 5 | Random Forest | 0.115405743 | 0.09 |
| 5 | SVM | 0.097540574 | 0.08 |
| 5 | XGB | 0.132209738 | 0.11 |
| 10 | Random Forest | 0.121048689 | 0.08 |
| 10 | SVM | 0.100848939 | 0.08 |
| 10 | XGB | 0.124444444 | 0.07 |
| 25 | Random Forest | 0.098689139 | 0.08 |
| 25 | SVM | 0.097490637 | 0.07 |
| 25 | XGB | 0.109875156 | 0.09 |
| 50 | Random Forest | 0.118838951 | 0.08 |
| 50 | SVM | 0.084107366 | 0.09 |
| 50 | XGB | 0.11772784 | 0.1 |
| 100 | Random Forest | 0.113283396 | 0.15 |
| 100 | SVM | 0.111011236 | 0.11 |
| 100 | XGB | 0.119975031 | 0.13 |

从表中发现随着主题数的增加，训练准确度和测试准确度在增加，因此可以初步判断，主题数的增加有利于分类准确性，XGB和随机森林的表现相对SVM较好。

Experiment Two:

在主题数 T 为 100，k 为 25，使用各个分类器的情况下分类性能的变化如下表所示

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| k | 主题 | 模型 | 类型 | 训练精度 | 测试精度 |
| 100 | 25 | Random Forest | Char | 0.269013733 | 0.35 |
| 100 | 25 | SVM | Char | 0.22525593 | 0.17 |
| 100 | 25 | XGB | Char | 0.281460674 | 0.29 |
| 100 | 25 | Random Forest | Word | 0.113258427 | 0.09 |
| 100 | 25 | SVM | Word | 0.095318352 | 0.11 |
| 100 | 25 | XGB | Word | 0.107652934 | 0.08 |

从表中可以看出，在Char和Word中，Random Forest的测试精度最高，并且以char为基本单元准确度较高。

Experiment Three:

在主题数 T 为 50，基本单元为word，使用各个分类器的情况下分类性能的如下表所示

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| K | 模型 | 类型 | 训练精度 | 测试精度 |
| 20 | Random Forest | Word | 0.126729089 | 0.08 |
| 20 | SVM | Word | 0.108751561 | 0.09 |
| 20 | XGB | Word | 0.114369538 | 0.09 |
| 20 | Random Forest | Word | 0.122259675 | 0.08 |
| 20 | SVM | Word | 0.108751561 | 0.09 |
| 20 | XGB | Word | 0.114369538 | 0.09 |
| 100 | Random Forest | Word | 0.111036205 | 0.09 |
| 100 | SVM | Word | 0.095318352 | 0.11 |
| 100 | XGB | Word | 0.107652934 | 0.08 |
| 100 | Random Forest | Word | 0.113258427 | 0.09 |
| 100 | SVM | Word | 0.095318352 | 0.11 |
| 100 | XGB | Word | 0.107652934 | 0.08 |
| 500 | Random Forest | Word | 0.09752809 | 0.12 |
| 500 | SVM | Word | 0.08855181 | 0.07 |
| 500 | XGB | Word | 0.09752809 | 0.15 |
| 1000 | Random Forest | Word | 0.104244694 | 0.12 |
| 1000 | SVM | Word | 0.080724095 | 0.09 |
| 1000 | XGB | Word | 0.109850187 | 0.11 |
| 3000 | Random Forest | Word | 0.098689139 | 0.08 |
| 3000 | SVM | Word | 0.097490637 | 0.07 |
| 3000 | XGB | Word | 0.109875156 | 0.09 |

从表中可以看出随着k 的增加，训练准确度和测试准确度先增后减小，因此 K 选择 500 最为合适。

**References**

[1] https://blog.csdn.net/Katherine\_Cai\_7/article/details/81634605

[2] https://blog.csdn.net/weixin\_44852067/article/details/130346159

[3] <https://www.zhihu.com/tardis/zm/art/31886934?source_id=1003>

[4] https://blog.csdn.net/weixin\_44852067/article/details/130346159