Data Mining 实验报告

Homework VSM and KNN

Homework NBC

Homework Clustering

学号：201834889

姓名：张玉卉

Homework VSM and KNN

一 实验要求：

1. 使用 github 管理项目
2. 建立向量空间模型来表示文本
3. 创建 KNN 分类器，用 KNN 实现文本分类

二 实验步骤

（一）VSM 建立向量空间模型来表示文本

1. 读取所有文档， 遍历文件夹
2. 对获取的文档进行处理 tokenizaton、normalization、stopwords、Stemming、punctuation_remove
3. 预处理遍历文件的同时计算词频（一个词在所有文档出现的次数） ， 并保存处理好的文档和单词词典
4. 过滤掉单词词典中词频太低和过高的单词
5. 计算词典中单词的 DF、 IDF
6. 生成文档向量 vectors， 为向量打标签表明其所属类别

（二）创建 KNN 分类器，用 KNN 实现文本分类

第一种方法（未使用类库）

1. 获取并使用 VSM 生成的向量
2. 将向量划分为测试和训练集合，随机选取 20%作为测试集，剩下 80%作为训练集。
3. 创建 KNN 分类器，进行 knn 分类： 采用余弦相似度进行计算文档相似度。计算测试文档与训练文档的余弦，选出 K 个距离最近的训练文档，统计排序 K 个文档中哪个类别最高，则该测试文档就为哪个类别。
4. 计算 KNN 分类的准确率，将计算出来的结果与真实值进行比较， 统计归类正确的文档数目除以总的测试集文档数目统计并输出正确率。
5. 调整不同的 K 值， 查看正确率变化

第二种方法（使用类库）

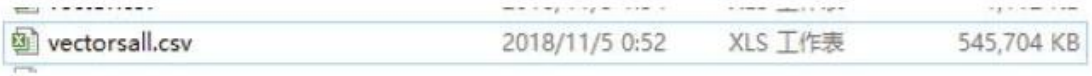
使用 sklearn neighbors KNeighborsClassifier 的分类器和 numpy

1. 读取数据， 分别以矩阵的形式读取训练集和测试集合
2. 读取训练集和测试集的标签
3. 获取类库中的 KNeighborsClassifier 分类器， 输入测试集和测试集标签进行训练
4. 预测测试集的标签
5. 比较预测值和真实值， 计算准确率

三 实验结果

1. VSM 生成了 5875 的词典和文档向量，向量保存为 CSV 格式，500 多 MB 无法上

传 github（100MBlimit） ， 所以随机上传了部分 vector（低于 100MB）



2. 正确率

第一种方法 KNN 的正确率如下表

k	Accuracy
1	0.72853
3	0.76708
6	0.73208
10	0.74666
20	0.69291
25	0.65208
50	0.62853

第二种方法 KNN 只跑了少量数据， K 值在 1-30 左右时 accuracy 大约在 0.69-0.78 左右

Homework NBC

一 实验要求

实现朴素贝叶斯分类器

测试其在 20 Newsgroups 数据集上的效果

二 实验步骤

1、 数据读取并且处理， 遍历文件夹读取全部数据

2、文本大写转小写、 分词、 词干还原、 去停用词、 去标点符号和与单词无关的字符

3、将每一篇处理好文档保存起来， 并且为文档建一个标签 list， 并生成词汇表、 处理文档同时记录每一个词汇出现过所有的次数

4、过滤出现次数低于 20 高于 800 的单词生成词典。

5、将保存好的文档划分为测试集和训练集， 比列为 20%和 80%

6、训练过程， 计算词频， 对存放词频向量的训练集进行训练， 首先计算先验概率： P(S)=某类文档数/文档总数。 其次每个类中每个词出现的概率： P(word|S)=(类 S 中 word 出现总数+1)/(类 S 中出现的单词总数+词典长度)。

7、 测试过程， 用朴素贝叶斯公式计算测试文档属于某个类的概率，哪个概率最大就属于哪个类， 最后依据测试集的标签统计正确率。

三实验结果

20news 中随机挑选 20%测试集实验输出的分类正确率为 86.7%

Homework Clustering

一 实验要求

1. 建立向量空间模型

2. 实现聚类算法

二 实验步骤

（一）建立向量空间模型

1. 读取 Tweets.json，并将结果单词向量存放在 vectors 中，将标签存放在 label 中。

2. 生成词汇表：遍历所有文档向量，即 vectors。将所有单词无重复的存入词汇表 wordtable 中。由于单词数比较少，所以不进行词频过滤。

3. 计算 tf-idf：循环遍历所有文档，对于其中的一篇文档： tf：首先，计算这篇文档的单词在这篇文档中出现的次数作为 tf； idf：首先计算每个单词出现在不同文档中的文档数，并利用公式 idf=log((N+1)/(df+1))表示 idf。其中 N 为文档总数。

4. 将计算好的 tf-idf 向量保存在文件中，并把类别号保存在 tf-idf 向量的最后一维

（二）实现聚类算法

1. 首先加载之前 vsm 计算好的 tf-idf 向量，将其保存在 vectors 中，类别保存在 label 中。

2. 导入 sklearn 中的聚类算法，包括 k-means、 AffinityPropagation、DBSCAN 等。

3. 对每个聚类算法，若需要指定 k 个类，则 k 取数据集类别总数。对每个聚类算法进行适配、预测样本标签，计算 NMI 并返回 NMI 最终结果。

三 实验结果

实验利用 sklearn 写好的几个聚类算法对数据集进行聚

类，并计算 NMI：

算法	结果
K-Means	0.76256
AffinityPropagation	0.75941
Mean-shift	0.51091
Spectral Clustering	0.50504
Ward hierarchical Clustering	0.79775
DBSCAN	0.09149
Gaussian Mixtures	0.81548