VSM and KNN

姓名：祝瑶佳　　学号：201834893

项目：为文本建立 VSM 并使用 KNN 进行文本分类

一、实验目标

1. 掌握文本预处理的方法(提取词干，去除停用词等)。
2. 计算每个文档的词频得到词频矩阵。
3. 计算 TF-IDF 值，计算权重，提取关键词。
4. 得到 VSM。
5. 处理文档得到训练集和测试集。
6. 计算欧式距离。
7. 计算 K 个最近邻并合并。
8. 测试 KNN 分类效果。

二、实验任务

1. 预处理文本数据集，并且得到每个文本的 VSM 表示
2. 实现 KNN 分类器，测试其在 20Newsgroups 上的效果。

三、实验数据

20 Newsgroups

四、实验步骤

1，文本处理

（1）分词

读取文档按空格分词，并且去掉符号。将文档划分成单词，并对单词做一些处理：大写字母变成小写字母，名词复数变单数，去掉停用词，各种时态和形式的动词变成原形，只保留英文词干部分。

（2）划分数据集

将数据集划分成训练集和测试集，其中训练集占 80%，测试集占 20%。

（3）创建词典

从训练集中读取所有的文档，统计所有的单词及词频，计算 TF-IDF 的值，提取关键词，并创建字典。

（4）得到文本的向量表示

1. 使用 KNN 文本分类

将数据集划分为训练集和数据集，计算每一个测试实例到训练集实例的欧式距离，对所有距离进行排序，得到 K 个最近邻。对最近邻进行合并排序，最后测试分类准确度。

五、实验结果

结果如下图：
数据集的词频矩阵：

```
d1 = [2, 0, 4, 3, 0, 1, 0, 2]
d2 = [0, 2, 4, 0, 2, 3, 0, 0]
d3 = [4, 0, 1, 3, 0, 1, 0, 1]
d4 = [0, 1, 0, 2, 0, 0, 1, 0]
d5 = [0, 0, 2, 0, 0, 4, 0, 0]
d6 = [1, 1, 0, 2, 0, 1, 1, 3]
d7 = [2, 1, 3, 4, 0, 2, 0, 2]
```

TF-IDF：

```
idf:
[0.24303804868629444, 0.24303804868629444, 0.146128035678238, 0.146128035678238, 0.8450980400142568, 0.066946789630
61322, 0.5440680443502757, 0.24303804868629444]
```

计算距离：

```
>>> distances  # 对应的距离
array([[ 13.37908816,  13.60147051,  13.60147051,  13.60147051,
         13.60147051,  13.6381817 ]])
```

六、实验结论和感想

1. 本实验中，k 值的选取不同，准确率也存在差别，可以多尝试下不同的 K 值，找到使得准确率最高的 K 值。
2. 不管是在 VSM 模型的建立中，还是 KNN 分类中，由于数据量很大，会发生内存溢出的现象。