山东大学 计算机科学与技术 学院

计算机视觉 课程实验报告

学号：201600130032	姓名： 陆禹承	班级： 智能 16
实验题目：OpenCV 配置及图像基本操作		
实验内容： 处理文档，用 VSM 模型表示		
实验过程中遇到和解决的问题： 使用 python 来编写程序 先观察数据，数据中包含很多杂七杂八的符号、网络用语、地址连接等等。 Token 阶段打算忽略掉 TAB、ENTER、逗号、句号，然后直接分段。 分完以后使用库进行词形还原，然后词干提取，全部存为名词。 去除停用词以及其他一些意义不明的词（夹杂着特殊符号） 编码问题全部忽视，通常英文文档中出现的应该是乱码 处理完词之后，先统计共有多少种词，然后每个文档一个向量，记录每个词出现的频率。		

处理用时比较短，最后的向量处理时间比较长。

一共 1.8w 文档，处理完后大约 2w 种单词。保留包含数字的项目，单词数翻倍。

期间使用了 textblob 进行 token，效果比自己单纯地 split 好很多，可以区分符号和单词混在一起的内容。

词形还原和词干提取也是调用库，因为英语中有很多不规则词，只能靠语料库实现。

拼写纠错没有使用，因为可能将一些网络用语或者其他词错误地纠正成英语单词（有的很有可能只是人名或网站名）因此没有使用。

如果需要使用的话，可以考虑对单词出现频率小的进行拼写纠错。

比如 accoount 这种拼写错误（统计中大概只出现一次）

实际矩阵中有超级多的 0，因为很多单词都没有在一个文档中出现。（存储的时候想压缩存储，因为实在太大了，感觉用链表存也比较省空间）

最后输出的矩阵有 8GB+，非常大，因此没有上传

原先想根据词频率来削减部分词，如果削除低频率词可能损失信息，削弱每个文档都出现

得差不多的词效果不明显(相对总词数这点微不足道)。最后选择妥协 ,看情况再削词数量。

最终输出字典+向量

TF-IDF 比较简单，只需要在现有数据上做一下统计即可。

（ DEBUG 过程，基本是 python 了解不够所致，比如元组问题、循环中不可更改，无隐式类型转换，编码问题）

结论分析与体会：

Python 基本入门，学会了基本文件操作和基本语法以及一些库的使用方法。

对文本预处理有了基本了解。