

实 验 报 告

学 号	20020006107	姓 名	王义钧	专业班级	通信工程	
课程名称	自然语言处理			学期	2022 年 春 季学期	
任课教师	仲国强	完成日期	2022/6/9	上机课时间	2022/6/9	
实 验 名 称	NLP 实验六：TF-IDF					

一、实验目的:

学习华为云课堂 TF-IDF 课程，输出重要词语的 TF-IDF 权重。

二、实验内容:

1. 完成华为云中自然语言处理理论、应用与实验课程中实验部分 4.1-4.4 的视频学习。保留学习后的截图。

2. 理解并学会 4.2 中的 TF-IDF，用代码实现 TF-IDF 的计算过程

三、实验过程:

1、华为云截图

华为云

华为云开发者学堂

学习路径 | 在线课程 | 云直播 | 实验室 | 华为云认证 | 培训服务 | 资讯

🔍 开发个人中心 ▾ | 文档 | 备案 | 控制台 | hid_0apccq7xw6... ▾

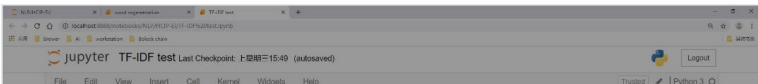
课程大纲

已学 47%

- 学前导读
- 第1章 自然语言处理简介
- 第2章 预备知识
- 第3章 关键技术及应用
- 第4章 自然语言处理实验
 - 4.1 中文文本分词
 - 4.2 TF-IDF特征处理
 - 4.3 Word Embedding: Word2Vec
 - 4.4 Sentences Embedding: Doc2Vec
 - 4.5 自然语言处理服务
 - 4.6 对话机器人服务
- 附：培训教材

TF-IDF特征处理

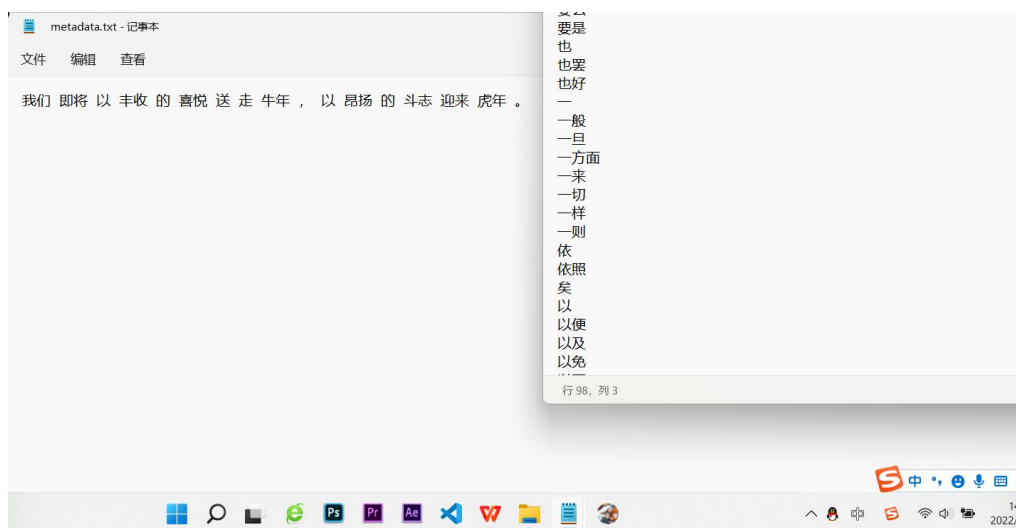
本视频主要讲解了配套实验手册中TF-IDF特征处理算法的实现操作。



2、TF-IDF 代码实现

首先规定好测试数据集与停顿词。根据视频，将测试数据集中每个分词后加上换行符，加入到停顿词字符串中。若这么做，会生成提示“标记停用词，而不在 `stop_words` 中生成标记['*', '*', '*', '*]'”。

解决方法是继续增加停用词的数量，根据 CSDN 上整理出来的一些去重在提取中文分词，加入到当前停顿词中。



然后将文本中的词语转换成词频矩阵，矩阵元素 $a[i][j]$ ，表示 j 词在 i 类文本下的词频。调用 sklearn 中的 **TfidfTransformer** 类，统计每个词语的 tf-idf 值。然后依次输出关键字及在文本中的位置，并将 tf-idf 抽取出来，0 元素 $a[i][j]$ 表示 j 词在 i 类文本中的 tf-idf 权重。最后将每个分词的权重输出出来。

```
vectorizer = CountVectorizer(stop_words = stopwords, min_df = 0) #该类会将文本
中的词语转换为词频矩阵，矩阵元素  $a[i][j]$ 。表示  $j$  词在  $i$  类文本下的词频
transformer = TfidfTransformer() #该类会统计每个词语的 tf-idf 权值
tfidf = transformer.fit_transform(vectorizer.fit_transform(dataset)) #第一个
fit_transform 是计算 tf-idf, 第二个 fit_transform 是将 文本转为词频矩阵
word = vectorizer.get_feature_names_out() #获取词袋模型中的所有词语
print("word:", word)
print(vectorizer.vocabulary_) #查看到所有文本的关键字和其位置

weight = tfidf.toarray() #将 tf-idf 矩阵抽取出来，0 元素  $a[i][j]$  表示  $j$  词在  $i$  类文本
中的 tf-idf 权重
print("weight:", weight)
```

四、 结果展示：

```
PS C:\Users\王义钧> python -u "d:\NLP\lab6\wvJNLP06.py"
1
['我们' '即将' '以' '丰收' '的喜悦' '送走' '牛年' '，' '以' '昂扬' '的' '斗志' '迎来' '虎年' '。']
也好'，'也罢'，'于是'，'于是乎'，'云云'，'他人'，'他们'，'以便'，'以免'，'以及'，'以至'，'以至于'，'以致'，'似的'，'依照'，'倘使'，'倘或'，'倘然'，'倘若'，'再者'，'再说'，'即将'，'只是'，'只有'，'只要'，'只限'，'同时'，'向着'，'呜呼'，'咱们'，'喜悦'，'嘻嘻'，'因为'，'因此'，'因而'，'在下'，'她们'，'它们'，'怎么'，'怎么办'，'怎么样'，'怎样'，'我们'，'所以'，'抑或'，'斗志'，'无宁'，'无论'，'昂扬'，'有些'，'有关'，'有的'，'正如'，'奥字'，'沿着'，'照着'，'牛年'，'由于'，'由此可见'，'相对而言'，'着呢'，'自个儿'，'自从'，'至于'，'虎年'，'虽则'，'虽然'，'虽说'，'要不'，'要不是'，'要不然'，'要么'，'要是'，'诸位'，'谁知'，'越是'，'迎来'，'这个'，'这么'，'这么些'，'这么样'，'这么点儿'，'这些'，'这会儿'，'这儿'，'这就是说'，'这时'，'这样'，'这边'，'这里'，'通过'，'随着'，'顺着'，'首先'] not in stop_words
.
% sorted(inconsistent)
word: ['丰收' '即将' '喜悦' '我们' '斗志' '昂扬' '牛年' '虎年' '迎来']
{'我们': 3, '即将': 1, '丰收': 0, '喜悦': 2, '牛年': 6, '昂扬': 5, '斗志': 4, '迎来': 8, '虎年': 7}
weight: [[0.33333333 0.33333333 0.33333333 0.33333333 0.33333333 0.33333333 0.33333333
0.33333333 0.33333333 0.33333333]]
-----这里输出第0类文本的词语tf-idf权重-----
丰收 0.3333333333333333
-----这里输出第1类文本的词语tf-idf权重-----
即将 0.3333333333333333
-----这里输出第2类文本的词语tf-idf权重-----
喜悦 0.3333333333333333
-----这里输出第3类文本的词语tf-idf权重-----
我们 0.3333333333333333
-----这里输出第4类文本的词语tf-idf权重-----
斗志 0.3333333333333333
-----这里输出第5类文本的词语tf-idf权重-----
昂扬 0.3333333333333333
-----这里输出第6类文本的词语tf-idf权重-----
牛年 0.3333333333333333
-----这里输出第7类文本的词语tf-idf权重-----
虎年 0.3333333333333333
-----这里输出第8类文本的词语tf-idf权重-----
迎来 0.3333333333333333
```

五、 心得体会：

通过华为云教程，手动实现复现代码，对计算 TF 与 IDF 的过程有了更加深入的了解，并结合在之前实验中学习到的分词知识，来帮助我完善代码功能，学会运用二维数组和权值等来表示 TF-IDF 权值。