第二天笔记-自然语言处理，分词技术，词性标注，文本处理

- 常见的NLP技术
  - 1.词法分析
    - 分词技术 （根据句子中正确的意思，把词组进行切分的技术）
      - 把句子分成不同的组词成分
    - 词性标注 （为分词的结果中的每一个单词标注一个正确的词性）
      - 有特殊词性的词，形容词，副词，语气词等
    - 命名实体识别-NER （文本中具有特定意义的实体）
      - 人名，地名，时间，特殊的词
      - 一般人物就是识别出待处理文本中的三大类和七小类
        - 三大类： 实体类，时间类 ， 数字类
        - 七小类： 人名，机构名，地名，时间，日期，货币，百分比
      - 步骤：
        - 1.实体边界识别
        - 2.确定实体类别
          - 1.英文实体
          - 2.中文实体
      - 如何进行命名实体识别
        - 1.基于规则和词典的方法 （每年都更新词，词库太大，规则太多，不好用）
        - 2.基于统计的方法
          - 1.隐马尔科夫模型-HMM -- 统计分析模型
            - 隐马尔可夫模型
              - 是结构最简单的动态贝叶斯网，著名的有向图模型，主要用于时序数据建模（语音识别，自然语言处理等）
              - 是马尔科夫链（马尔科夫过程的离散状态）的一种，状态不能直接被观察到，但能通过观测向量序列，每个观测向量都是通过某些概率密度分布表现为各种状态，每一个观测向量是由一个具有相应概率密度分布的状态序列产生
            - 隐马尔可夫模型构成五要素： 输入： 两个状态集合（N、M）三个概率矩阵（A、B、π） 输出： M中元素的概率值
              - N： 表示模型中的状态数，状态之间可以相互转移
              - M: 表示每个状态不同的观测符号，即输出字符的个数
              - A: 状态转移概率分布
              - B: 观察符号在各个状态下的概率分布
              - π: 表示初始状态分布
              - 应用场景 ：
                - 中文分词
                - 机器翻译
                - 语音识别

                - 通信中的译码
            - 马尔科夫过程
                - 是一类随机的过程在已知目前状态的条件下，它未来的演变不依赖于它以往的演变，通过对不同状态的初始概率以及状态之间的转移概率的研究，来确定状态的变化趋势
        - 2.较大熵 -ME
        - 3.支持向量机 - SVM
        - 4.条件随机场 - CRF
    - 贝叶斯分类
        - 利用概率统计知识进行分类的算法，先验概率，后验概率
        - 朴素贝叶斯法NB是基于贝叶斯定理与特征条件独立假设的分类方法，最为广泛的两种分类模型时 ： 决策树模型和朴素贝叶斯模型
        - 应用情景
            - 1.文本分类
            - 2.垃圾邮件过滤
            - 3.多分类实时预测
            - 4.拼写纠错
- 词义消歧
- 2.句法分析
- 3.语义分析
-
- 构建一个词分类器
    - 第一步： 预处理数据 ， 让测试数据和训练数据进行均和
    - 第二步： 处理好的数据，利用CountVectorizer进行词向量转化
    - 第三步： 转化成词向量的数据，交给 TFIDF 进行数据特征的提取，变成最终的训练数据集
    - 第四步： 训练分类器，把数据带入分类器进行测试训练
    - 第五步： 循环重复过程，直到得到你想要的最优的结果
- vectorizer=CountVectorizer()#构建一个计算词频（TF）的玩意儿，当然这里面不只是可以做这些transformer=TfidfTransformer()#构建一个计算TF-IDF的玩意儿 tfidf=transformer.fit_transform(vectorizer.fit_transform(corpus))
- #vectorizer.fit_transform(corpus)将文本corpus输入，得到词频矩阵#将这个矩阵作为输入，用transformer.fit_transform(词频矩阵)得到TF-IDF权重矩阵
-
- 特征提取的方法
    - TF-IDF及其算法
        - 定义：
            - 是一种用于资讯检索与资讯探勘的常用加权技术，TF-IDF是一种统计方法，用以评估一字词对于一个文件集或一个语料库中的其中一份文件的重要程度。字词的重要性随着它在文件中出现的次数成正比增加，但同时会随着它在语料库中出现的频率成反比下降。TF-IDF加权的各种形式常被搜寻引擎应用，作为文件与用户查询之间相关程度的度量或评级
        - 原理：

        - 在一份给定的文件里，词频 (term frequency, TF) 指的是某一个给定的词语在该文件中出现的次数。这个数字通常会被归一化（分子一般小于分母 区别于IDF），以防止它偏向长的文件。（同一个词语在长文件里可能会比短文件有更高的词频，而不管该词语重要与否。）
        - 逆向文件频率 (inverse document frequency, IDF) 是一个词语普遍重要性的度量。某一特定词语的IDF，可以由总文件数目除以包含该词语之文件的数目，再将得到的商取对数得到。
        - 某一特定文件内的高词语频率，以及该词语在整个文件集合中的低文件频率，可以产生出高权重的TF-IDF。因此，TF-IDF倾向于过滤掉常见的词语，保留重要的词语。
    - 举例：
        - 词频 (TF) 是一词语出现的次数除以该文件的总词语数。假如一篇文件的总词语数是100个，而词语“母牛”出现了3次，那么“母牛”一词在该文件中的词频就是3/100=0.03。一个计算文件频率 (DF) 的方法是测定有多少份文件出现过“母牛”一词，然后除以文件集里包含的文件总数。所以，如果“母牛”一词在1,000份文件出现过，而文件总数是10,000,000份的话，其逆向文件频率就是 log(10,000,000 / 1,000)=4。最后的TF-IDF的分数为0.03 * 4=0.12
-
- 转化成词向量的方法 - 词频的意思
    - 基于词频数的文档向量 CountVectorizer ，是一种数据的抽象处理
    - 基于上下文语境来获取的词向量 word2vec
        - 是一群用来产生词向量的相关模型，这些模型为浅而双层的神经网络，用来训练以重新建构语言学之词文本
        - 一般有两种模型：
            - CBOW：
                - 模型是由： 输入层，映射层，输出层共同构成
                - 是一个二叉树结构，这个结构应用到word2vec中称之为Hierarchical Softmax
                - 输出是词向量
            - Skip-gram
                - 模型是由： 输入层，映射层，输出层共同构成
                - 是一个二叉树结构
                - 输入是一个特定的词向量，和CBOW正好相反
                - 输出是特定词的上下文词向量
-
- 常用的分词库
    - jieba
        - 结巴的三种分词模式
            - 全模式 seg_list = jieba.cut("我来到北京清华大学", cut_all=True)
            - 精确模式（默认模式） seg_list = jieba.cut("我来到北京清华大学", cut_all=False)
            - 搜索引擎模式 seg_list = jieba.cut_for_search("小明硕士毕业于中国科学院计算所，后在日本京都大学深造")
-

-
- 语料库
  - 什么是语料库?
    - 语料： 即语言材料，是语言学研究的内容，是构成语料库的基本单元
    - 语料库中存放的是在语言的实际使用中真实出现过的语言材料
    - 语料库是以电子计算机为载体承载语言知识的基础资源
    - 真实语料需要经过加工（分析和处理）才能成为有用的资源
  - 语料库的种类?
    - 异质的： 语料不同种类的都有
    - 同质的： 语言类型是都一个类型的，能形成某种规律，归为一类的
    - 系统的： 针对同一种的类型，财经啊，小说啊，等等
    - 专用的：
  - 语料的获取途径?
    - 爬虫
    - 公开数据集
      - 中科院自动化所的中英文新闻语料库
      - 搜狗的中文新闻语料库
      - 人工生成的机器阅读理解数据集-微软
      - 一个开放问题与回答的挑战数据集-微软
    - 自由平台，公司自己的
  - 语料处理步骤？
    - 获取语料
    - 格式化文本
    - 特征工程
-
- NLP的语言模型
  - 定义： 判断一个句子中的词的排序概率，或者或是句子是否是正确，通顺的概率
  - 概率语言模型两种类型： （预测字符串的概率，动机，如何计算）
    - Unigram models （一元文本法统计模型，是N元的一种特殊情况）
      - 成立的前提是条件无关： 计算方法是：每个字的概率相乘得到的结果
    - N - gram 语言模型 （N元模型）
      - N一般不会超过3，大于3时基本就无法出咯，参数空间太大，另外它不能表示词与词之间的关联性
-
- 文本处理方法步骤？
  - 第一步： 数据清洗（去掉无意义的标签，url，符号等）
    - url： 广告，循环连接，无效连接等等
    - 符号： 连接符号，表情符号
  - 第二步： 分词，大小写转换，添加句首句尾，词性标注
    - 添加句首句尾： 告诉聊天机器人，聊天的开始和结束
    - 大小写转换： 有些大写不用转换，例如 China
    - 词性标注： 词性，形容词，副词，名词，等等

- 第三步： 统计词频，抽取文本特征，特征选择，计算特征权重，归一化
- 第四步： 划分数据，分为训练集，测试集