# 自然语言处理第一次实验

### 实验目的

* 了解分词与词向量的训练。

分词就是将句子、段落、文章这种长文本，分解为以字词为单位的数据结构，方便后续的处理分析工作。词向量，顾名思义就是⽤来表⽰词的向量，也可被认为是词的特征向量或表征。把词映射为实数域向量的技术也叫词嵌入。我们通常在训练语言模型的同时得到词向量。

* 了解Word2Vec词向量的特性。

词向量本质是将一些低维、离散、不带任何意义的序号映射成带有特定任务性质的高维特征。为了研究这些特性，可以显式地通过余弦距离来比较两个词的相似性，也可以通过词间的类比关系来探查词向量的影藏特性，另外还可以通过PCA降维的方式将词向量可视化。

### 实验环境

python 3 + jieba + gensim + sklearn + matplotlib + NumPy + seaborn

* python3

除了高性能外，拥有NumPy、SciPy等优秀的数值计算、统计分析库。TensorFlow、Caffe等著名的深度学习框架都提供了Python接口。

* jieba

jieba是一款优秀的Python第三方中文分词库，支持三种分词模式：精确模式、全模式和搜索引擎模式。

* gensim

gensim是一款开源的第三方Python工具包，用于从原始的非结构化的文本中，无监督地学习到文本隐层的向量表达。它支持包括TF-IDF，LSA，LDA，和word2vec在内的多种算法。

* Sklearn

Sklearn (全称 Scikit-Learn) 是基于 Python 语言的机器学习工具。它建立在 NumPy, SciPy, Pandas 和 Matplotlib 之上，Sklearn 里面有六大任务模块：分别是分类、回归、聚类、降维、模型选择和预处理。

* Matplotlib

Matplotlib 是 Python 的绘图库，可与 NumPy 一起使用，提供了一种有效的 MATLAB 开源替代方案。

* NumPy

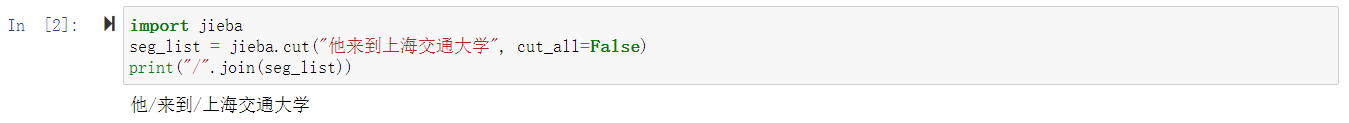
NumPy是Python语言的一个扩展程序库，支持大量的维度数组与矩阵运算，此外也针对数组运算提供大量的数学函数库。

* Seaborn

Seaborn是基于Matplotlib的图形可视化Python包，在Matplotlib的基础上进行了更高级的封装，使得作图更加容易。

### 实验步骤

1. 使用jieba分词工具进行分词，使用方法：jieba.cut(text)；

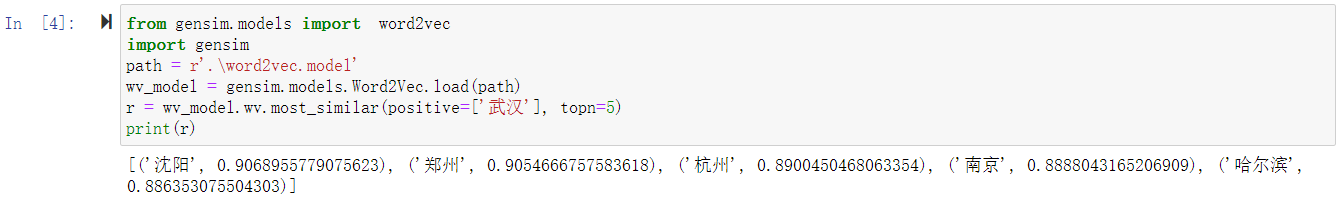


2. 使用gensim中的Word2Vec模型训练词向量：model = Word2Vec(common\_texts, size=100, window=5, min\_count=1, workers=4)；min\_count指定了需要训练词语的最小出现次数，默认为5；size指定了训练时词向量维度，默认为100；worker指定了完成训练过程的线程数，默认为1不使用多线程。

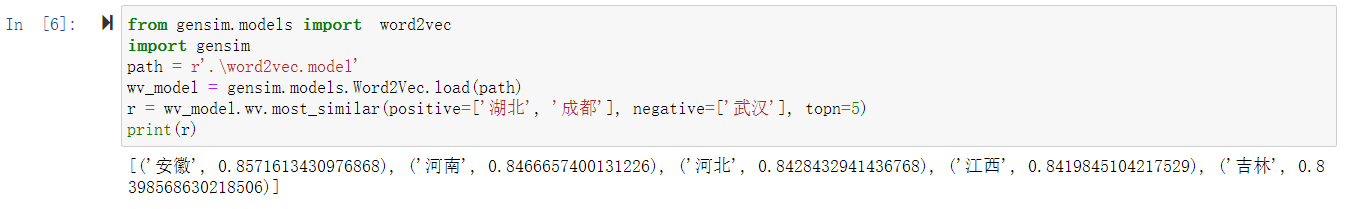
3. 使用训练好的词向量对指定的词（2个例子）进行相关性比较：model.similarity('中国','中华')；



4. 使用训练好的词向量选出与指定词（2个例子）最相似的5个词：model.wv.most\_similar(positive=['武汉'], topn=5)；

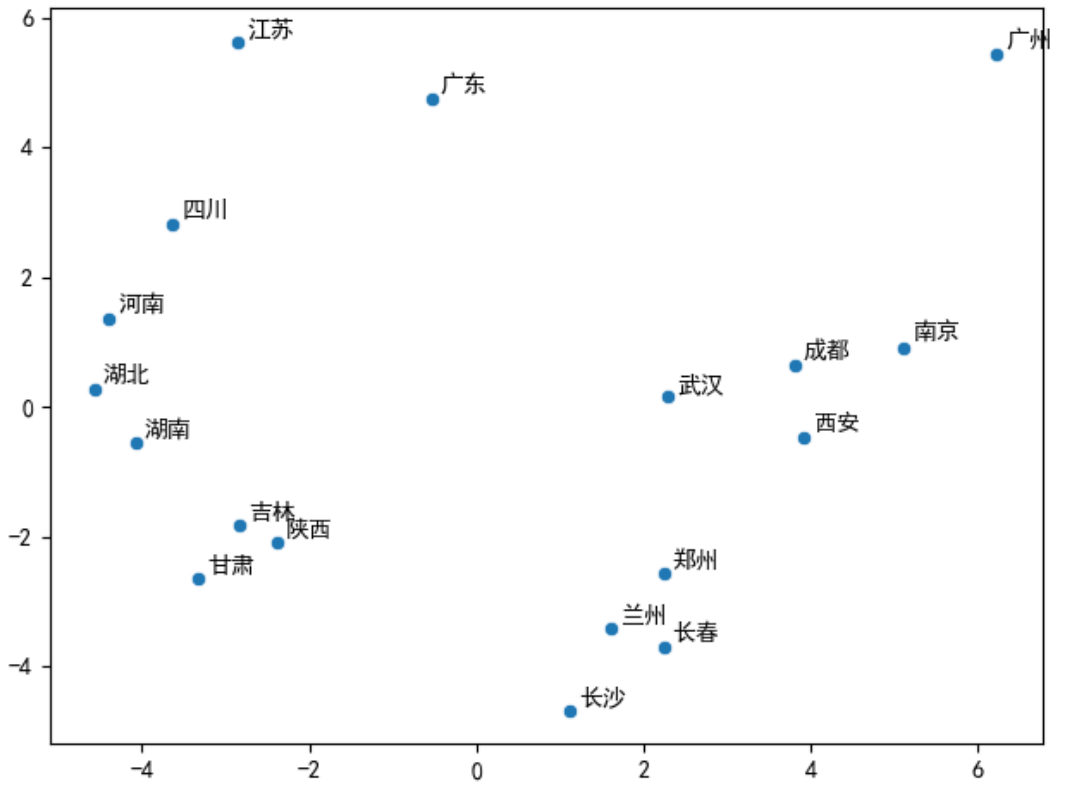


5. 使用训练好的词向量选出与指定词类比最相似的5个词（2个例子），如**湖北 - 武汉 + 成都 = 四川**： model.wv.most\_similar(positive=['湖北', '成都'], negative=['武汉'], topn=5)；



6. 使用sklearn中的PCA方法对列表['江苏', '南京', '成都', '四川', '湖北', '武汉', '河南', '郑州', '甘肃', '兰州', '湖南', '长沙', '陕西', '西安', '吉林', '长春', '广东', '广州', '浙江', '杭州']（可换成其他）中的所有词的词向量进行降维并使用seaborn和matplotlib将其可视化：

|  |
| --- |
| pca = PCA(n\_components=2) |
| results = pca.fit\_transform(embeddings) |
| sns.scatterplot(x=results[:, 0], y=results[:, 1]) |



### 提交时间

**10月23号截止**

各班班长或学习委员收集班内所有同学的实验报告和实验代码后发送到3263979158@qq.com邮箱中。

将所有文件打包为zip格式的压缩包，命名方式：姓名-学号-第1次实验

### 实验要求

* 完成所有实验内容
* 良好的代码风格
* 完整的实验报告

### 参考资料

1. [jieba文档](https://github.com/fxsjy/jieba)

2. [Word2Vec Model in gensim](https://radimrehurek.com/gensim/auto_examples/tutorials/run_word2vec.html#sphx-glr-auto-examples-tutorials-run-word2vec-py)

3. [PCA in sklearn](https://scikit-learn.org/stable/modules/generated/sklearn.decomposition.PCA.html)

4. [Word2Vec中的数学原理详解](https://www.cnblogs.com/peghoty/p/3857839.html)

5. [Distributed Representations of Words and Phrases and their Compositionality](https://papers.nips.cc/paper/5021-distributed-representations-of-words-and-phrases-and-their-compositionality.pdf)

6. [Efficient Estimation of Word Representations in Vector Space](https://arxiv.org/pdf/1301.3781.pdf)

**附件1：**

**武汉大学国家网络安全学院**

**实验报告**

**课程名称 自然语言处理**

**专业年级 2019**

**姓 名**

**学 号**

**协 作 者 无**

**实验学期 2021-2022　 学年 第一 学期**

**课堂时数 课外时数**

**填写时间 2021 年 10 月 16 日**

|  |
| --- |
| **实验介绍** |
| **【实验名称】：自然语言处理第一次实验** |
| **【实验目的】：**   1. 了解分词与词向量的训练 2. 了解Word2Vec词向量的特性   **【实验环境】：**   1. Windows 10 家庭中文版 2. Anaconda Navigator 1.9.12 3. conda 4.8.3 4. Python 3.8.11 5. jieba 0.42.1 6. genism 3.8.3 7. sklearn 1.0 8. seaborn 0.11.2 9. matplotlib 3.4.3   **【参考文献】：**  **[1]** Embedding和Word2Vec实战  <https://www.cnblogs.com/dogecheng/p/11565530.html>  **[2]** 通俗易懂地讲解什么是 PCA  https://www.zhihu.com/question/41120789 |
| **实验内容** |
| **【实验方案设计】：**  实验模型：   1. Word2Vec   词向量的嵌入在自然语言处理领域中是一个非常重要的课题，主要作用是将词语用向量来表示。在embedding中，独热码(one-hot code)也是一种可以尝试使用的编码方式。然而，这种方法存在一大缺点，即当数据量十分庞大时使用独热码几乎是不可能的，因为独热码占据了过多的空间以及过高的维度。这时候就能够体现出word2vec这一词嵌入方法的优势所在，它可以将更多的信息塞入更低的维度中。word2vec会通过数据集中的所有词语来构建一个词语字典，在这个字典中每个词语都有一个对应的索引值，因此一个句子就可以用这些索引值来表示，这样就实现了中文到数字的初步转换。图1对embedding进行了一个简单说明。  图1 embedding的实现步骤[1]  图1形象地说明了如何将句子中的词语用数字来进行表示，同时不丢失其原意。然后需要将已经转换为数字表示的句子嵌入到embedding层中，该层需一般要样本数和嵌入维度两个参数，样本数即上述构建的词语字典的大小，嵌入维度是自定义的。  图2 embedding的实现步骤[1]   1. PCA   PCA是一种常用的数据分析方式，功能是对数据进行降维处理，可以提取数据的主要特征分量。  以一个形象的例子进行说明，一般需要两个值来表示一个二维数据，而二维数据点可以在二维坐标系上表示出来。当情况最为简单时，这些样本点可以连成一条直线，那么旋转坐标轴，使其中一条坐标轴与该直线重合，此时就可以只用一个数据来表示其中任何一个样本点，二维数据降为了一维数据，成功实现了数据的降维。如图3所示，这是初始的二维数据样本点，假设这些样本点刚好可以连成一条直线。    图3 降维前的数据[2]  如下图4所示，通过旋转坐标轴使得原来的横轴与样本点连成的直线重合，此时只需要一个值来表示每个样本点，当然这个值需要重新进行计算。    图4 降维后的数据[2]  实际情况当然会更为复杂，数据不一定是二维，即使是二维一般样本点也难以连成一条直线。不过这是降维思想的基础，PCA能够在丢失尽可能少信息的情况下，将数据降维。  实验步骤：   1. 使用jieba进行分词     其中，jieba.cut函数的必要参数是一个待分词的字符串，cut\_all默认为False，表示使用精确模式进行分词，若为True表示使用全模式。如下图所示，jieba支持四种分词模式。     1. 使用gensim中的Word2Vec模型训练词向量   在构建模型之前，首先需要对数据集进行分词处理，然后使用word2vec.LineSentence将数据集中的每一行当作一个句子。    然后使用得到的sentences进行word2vec，其中参数size表示词向量维度、window表示训练窗口大小，即在训练一个词时需要考虑前后词的数量、min\_count表示需要训练的词语的最小出现次数、workers表示完成训练的线程数。训练完成后将该模型保存在本地，需要使用的时候可以直接读取模型文件，而不需要重新进行训练。     1. 使用训练好的词向量对指定的词进行相关性比较   使用model.wv.similarity对指定的词的相关性进行比较。在gensim更早的版本使用的是model.similarity方法。     1. 使用训练好的词向量选出与指定词类比最相似的5个词   首先通过本地保存的模型文件以导入模型，然后使用model.wv.most\_similar找出与指定词最相似的词语，其中positive表示结果与该词需要尽量相像，呈正相关。如下图所示，该方法所找出的词与“武汉”一样都是城市。     1. 使用训练好的词向量选出与指定词类比最相似的5个词   首先仍需要导入模型，然后仍使用model.wv.most\_similar找词，其中negative表示结果与该词需要尽量不相像，呈负相关。如下图所示，随着positive和negative参数值的改变，结果也从城市类的词语转变为了省份类的词语。     1. 使用sklearn中的PCA方法对指定列表中的所有词的词向量进行降维并使用seaborn和matplotlib将其可视化   首先导入模型，然后得到指定列表中的所有词的词向量，并将其构造为二维数组形式，以符合pca.fit\_transform所要求的输入维度格式。随后构建PCA模型，参数n\_components表示模型最终保存下来的特征数。最后使用seaborn和matplotlib绘制散点图，并且使用plt.annotate在每个点旁边添加中文注释，以表示每个点所对应的词语。    **【实验结果分析】：**  将列表中所有词的降维后的词向量进行可视化，结果如下图所示。 |
| **【实验总结】：** |
| **评语及评分（指导教师）** |
| **【评语】：**      **评分：**  **日期：** |

**附件2：**

**实验报告说明**

**1．实验名称：**要用最简练的语言反映实验的内容。

**2．实验目的**：目的要明确，要抓住重点。

**3．实验环境**：实验用的软硬件环境（配置）。

**4．实验方案设计（思路、**步骤和方法等**）**：这是实验报告极其重要的内容。包括概要设计、详细设计和核心算法说明及分析，系统开发工具等。应同时提交程序或设计电子版。

对于**设计型和综合型实验**，在上述内容基础上还应该画出流程图、设计思路和设计方法，再配以相应的文字说明。

对于**创新型实验**，还应注明其创新点、特色。

**5．实验结果分析：**即根据实验过程中所见到的现象和测得的数据，进行对比分析并做出结论（可以将部分测试结果进行截屏）。

**6．实验总结：**对本次实验的心得体会，所遇到的问题及解决方法，其他思考和建议。

**7．评语及评分：**指导教师依据学生的实际报告内容，用简练语言给出本次实验报告的评价和价值。