# 实验6《词嵌入模型》

实验学时： 2 实验地点： 二综204 实验日期： 2019/11/25

1. **问题描述**

文本是最常用的序列数据之一，可以理解为字符序列或单词序列，但是最常见的是单词级处理。本实验将会介绍单词和字符的one-hot编码，然后学会用Keras完成单词级的one-hot编码，然后介绍目前自然语言处理常用到的预处理方法词嵌入，并用Keras自带数据集完成简单的词嵌入模型。

1. **设计简要描述**

**1．**One-hot 编码

#### 1.1 one-hot编码简介

在最开始的实验2和实验3中，我们已经使用过简单的one-hot编码，one-hot编码是将标记转换为向量的最常用，最基本的方法。对于单词级别的one-hot编码，它将每个单词与一个唯一的整数索引相关联，然后将这个整数索引i转换为长度为N的二进制向量（N是词表的大小），这个向量只有第i个元素是1，其余都是0，对于字符级别的one-hot编码，它将每个字符与一个唯一的整数索引相关联，其余与单词级别的类似。

#### 1.2 单词级的one-hot编码

初始数据：每个样本是列表的一个元素（本例中的样本是一个句子，但也可以是一篇文档）

#### 1.3自行完成以下任务

1）给上述samples列表新增一个元素‘a panda is sleeping’

2) 自行完成单词级别的分词任务，并查看索引字典和样本列表的第二个元素的第二个单词的编码情况。（注意索引是0开始）

#### 1.4 字符级的one-hot编码

与上面类似现在我们通过以下方式来完成单词级的one-hot编码

#### 1.5 优化编码

请学员使用 1.3中新增的数据一个元素的数据集来重构索引字典并查看第三个元素中的第三个字符的编码情况

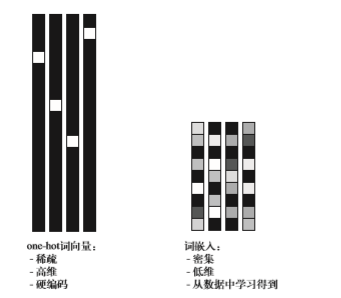
#### 1.6 用keras实现单词级的one-hot编码

我们可以通过以下简单的方式用keras来实现one-hot编码

2.词嵌入模型

#### 2.1 词嵌入简介

将单词与向量相关联的另外一个强大的方法就是词嵌入，one-hot编码得到的向量是二进制的、稀疏的（绝大部分你元素都是0）、维度很高（维度大小等于词表中的单词个数），而词嵌入是低微的浮点数向量（即密集向量，与稀疏向量相对），如下图所示，与one-hot编码得到的词向量不同，词嵌入是从数据中学习得到的。常见的词向量维度是256、512或1024（处理非常大的词表时）。与此相对，one-hot编码的词向量维度通常是20000或者更高（对应包含20000个标记的词表），因此词向量可以将更多的信息塞入更低的维度中。



#### 2.2 利用Embedding层学习词嵌入

#### 2.3 实例化Embedding层

可以将Embedding层理解为一个字典，将整数索引（表示特定单词）映射为密集向量。它接收整数作为输入，并在内部字典查找这些整数，然后返回相关联的向量。

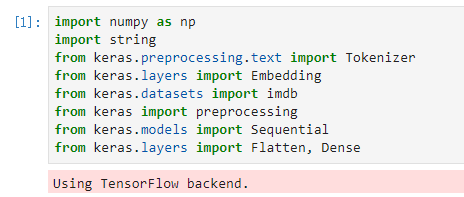
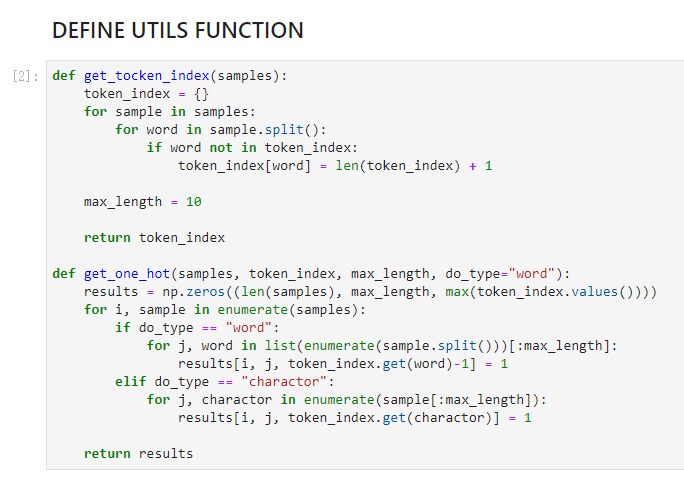


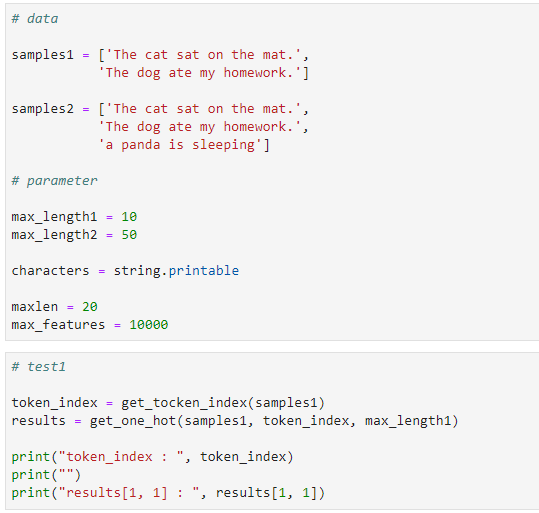
在我们通过以下方式来实例化一个Embedding

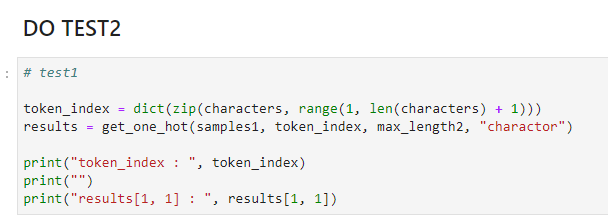
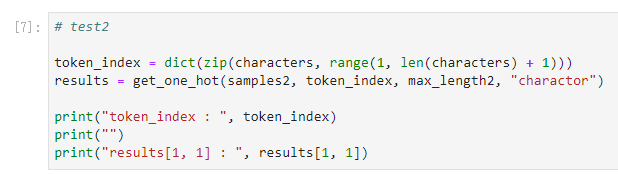
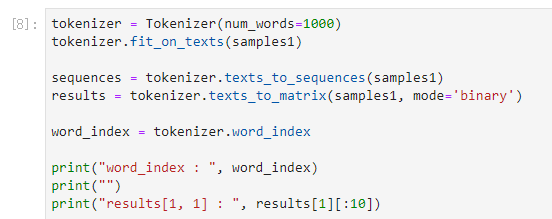
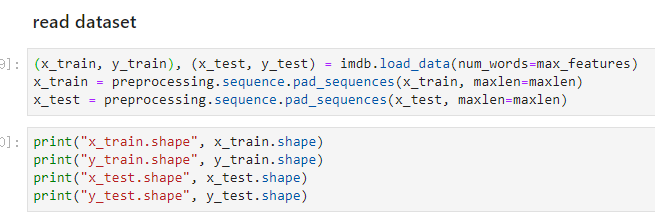
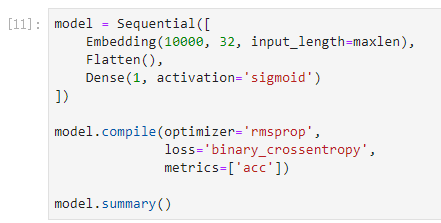
#### 2.4 加载IMDB数据

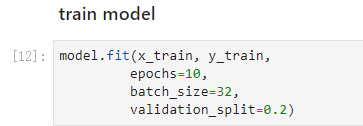
本节使用IMDB数据集，它包含来自互联网电影数据库（IMDB）的50000条严重两极分化的评论。数据集被分为用于训练的25000条评论和用于测试的25000条评论，训练集和测试集都包含50%的正面评论和50%的负面评论。

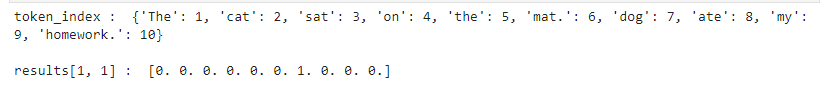
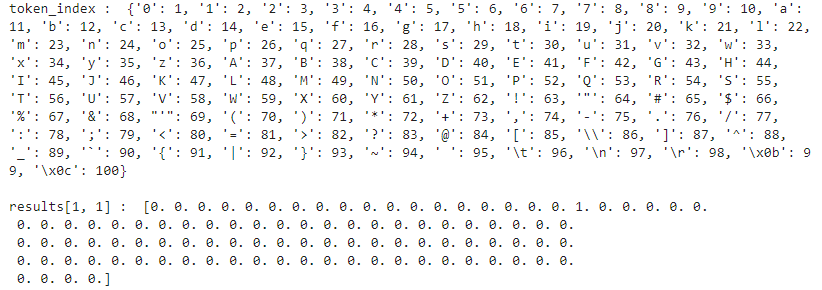
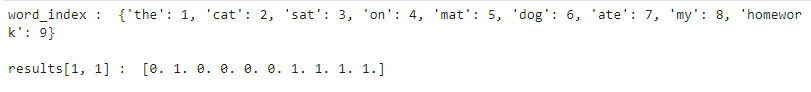
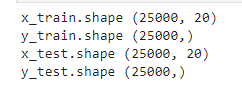
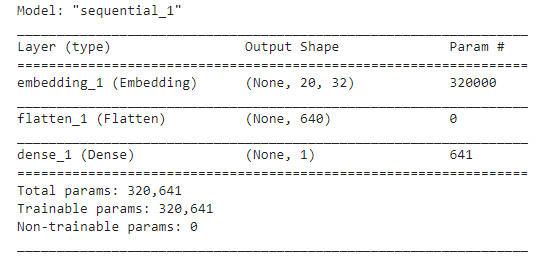
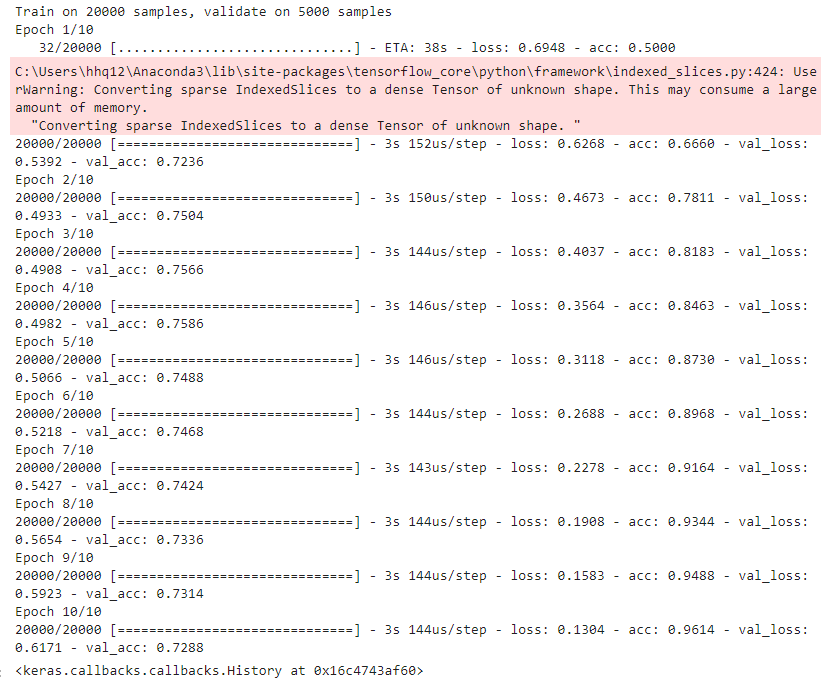
#### 2.5 构建keras模型并训练拟合模型

1. 导入包Sequential,Flatten,Dense,Embedding
2. 定义一个序列模型
3. 添加一个Embedding层，标记个数10000，维度8，输入长度是maxlen
4. 添加一个Flatten层
5. 添加一个全连接层，输出维度是1，激活函数‘sigmoid’
6. 编译模型，优化器选取‘rmsprop’，损失函数选取‘binary\_crossentropy’,评估方式是‘acc’
7. 用.summary()方法查看模型架构
8. 拟合模型，epoch选取10，batch\_size选取32，validation\_split为0.2
9. **程序清单**
10. **导入相应工具包**
11. **定义用来进行one-hot 编码的函数**
12. **单词的one-hot编码**



1. **字符级别的one-hot编码**
2. **使用keras进行one-hot编码**
3. **导入数据集**
4. **定义模型**
5. **模型训练**



1. **结果分析**
2. **导入相应工具包**
3. **定义用来进行one-hot 编码的函数**
4. **单词的one-hot编码**
5. **字符级别的one-hot编码**
6. **使用keras进行one-hot编码**
7. **导入数据集**
8. **定义模型**
9. **模型训练**
10. **调试报告**
11. **所遇到的问题**

本次实验是按照老师提供的代码一步一步来的，基本上没遇到问题。遇到了一些小问题，但很快就解决了。

1. **经验和体会**

感觉one-hot编码并不是太复杂吧，以前都是直接调函数的，实现起来也就几个循环。不管怎么说，自己又了解了一些关于编码和词嵌入模型的一些知识。感谢老师所提供的代码示例，让自己少走了许多弯路。

源码链接：

<https://github.com/sunnyswag/ml_dl_assignments/blob/master/assignment_6/assignemnt_6.ipynb>