**数据分析训练题**

**实验报告**

****

**学号：201600301079**

**姓名：崔玉峰**

**班级：2016级**

## 问题描述

1. **实验题目**

(实体识别)

现有一个Amazon的众多商品的数据记录文件(Amazon.csv)，同时有Google对众多商品的数据库记录文件(Google.csv)。如何将两者的数据匹配起来？

1. **解题思路**
   1. **分割文本，得到有效Tokens**

对于文本匹配，需要求出token "token" 其实就是指的一个不区分大小写单词，或者说是以空格等等的符号分割的一个个字符串。"tokens"则是token的列表。一个token允许在tokens中存在多次

* 1. **求解TF-IDF**

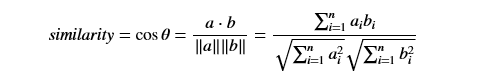
TF(token)=该token在tokens中的出现次数/tokens中的总token数

IDF(token)=tokens(token列表)的个数/出现过该token的tokens(token列表)的个数

TF-IDF:TF-IDF(token)=TF(token)\*IDF(token)

* 1. **三.求余弦相似度**

把每一个token理解成一个维度，那么对于一条Amazon的数据和一条Google的 数据，我们就可以计算二者TF-IDF的余弦值。可以把余弦值看作一种相似度。余弦值 越大就意味着二者越相似。



* 1. **与最优匹配比较，求出最佳阈值**

## 具体实现：

1. **加载数据**



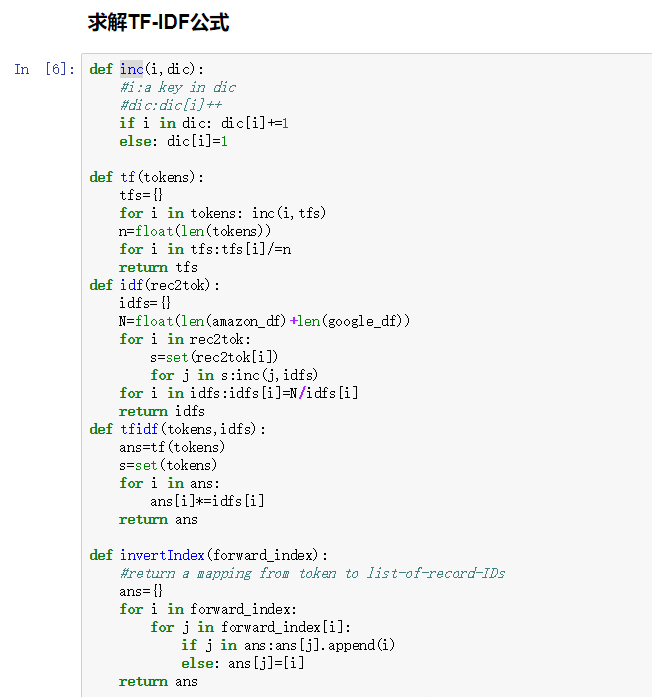
将给定的数据通过pandas库读进来

1. **分割tokens方法编写**

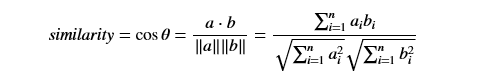


1. 通过正则表达式re库，进行tokens分割
2. 去除of,and,等无效tokens
3. 逆向索引方便后期优化算法，
4. **计算TF-IDF公式：**

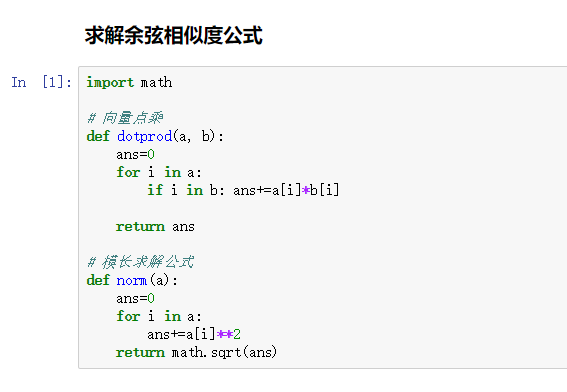
根据TF，IDF公式编写求解函数



1. **计算余弦相似度公式编写**



求解相似度公式 ，编写求解点乘公式和模长公式的方法：

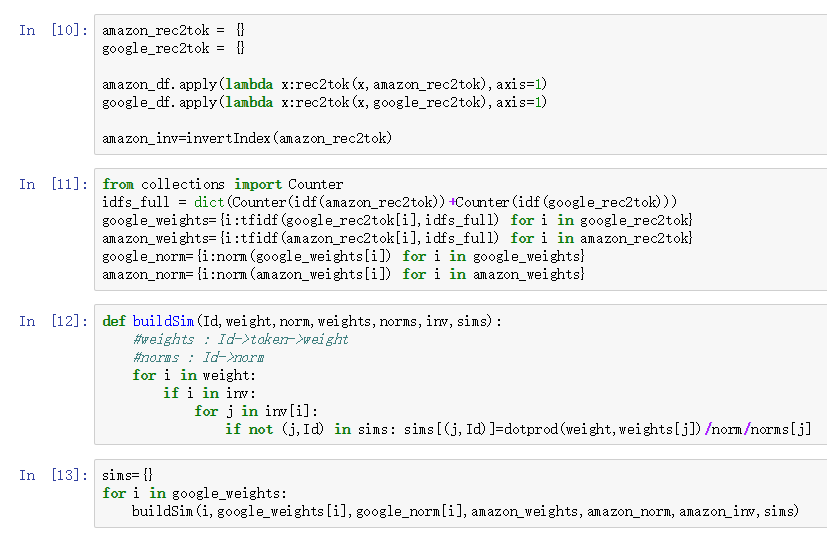


1. **计算匹配精确度方法**



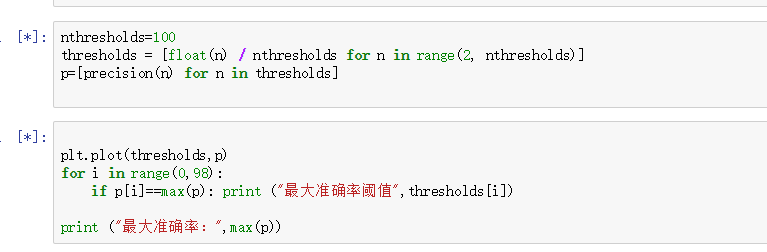
计算与最优匹配的精确度的相关方法，正确和错误的比例

1. **调用方法求解每条数据的余弦相似度**

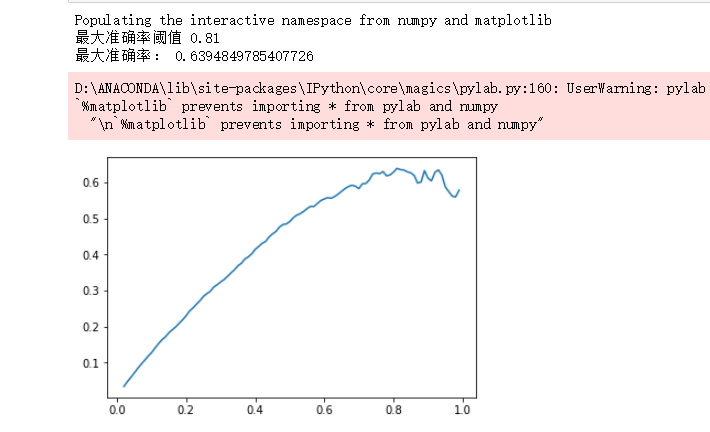


通过优化算法预处理出逆向索引与norms这两个字典，然后快速求出每条数据的余弦精确度

1. **将求出的结果与最优匹配比较并求出阈值：**



1. **输出结果：**



## 实验结论和感悟：

曾经在学习数据科学之前，接触过机器学习的相关算法，逻辑回归，神经网络等，但是一直有一个疑惑对于，因为这些算法都是经过数学推导的，好像只能处理数字，那该如何处理文本数据一直有疑惑，通过此次实验了解了如何处理文本数据，实体识别匹配。

因为对于python的代码语法不是很熟悉，所以很多的地方是按照老师所给出的模板代码敲出来的但是最后结果还是很令人满意的。