# 数据科学基础大作业

### 图文融合之虚假新闻

**组长: 刘怡然231250136 组员: 刘佳昱231250051, 周亦余 231250040**

*项目背景: 如今，随着社交软件和数字平台的普及，我们在日常生活中接触到的信息类型越来越多样化：文字、图片、视频等各种形式的信息充斥在我们的视野中。这种信息多元化的传播方式不仅仅提供了更多的信息内容，更使得人们的想法得到更全面准确的表达和传播。*

*然而，在人们享受复杂信息带来的多元交流之时，谣言的传播也在各个社交媒体上肆意横行，识别和屏蔽谣言的工作变得尤为重要。不同于以往单纯的“文字-->谣言”的简单对应关系，如今的“图片、视频、文字...-->谣言”的复杂对应关系需要我们采用新的分析方法，即图文融合。*

*对此，我们小组以微博提供的谣言汇总作为数据集，使用图文融合的方析方法对其进行分析，尝试获取谣言的基本特征以帮助识别和屏蔽谣言的工作。*

### 项目简介

本项目旨在运用统计与机器学习方法对虚假新闻的分辨能力进行提升。

对于一条新闻，在进行数据规范化后，可以通过两种模型对新闻文本做处理，可以获得判断该新闻真假的准确度等相关值，而后通过图文融合计算该新闻的图文相似度，再在真新闻和假新闻的两种假设前提下，通过计算置信区间获得判断真假的准确率，最后将两个准确率进行整合，一定程度上来提高分辨虚假新闻的能力。

### 数据获取

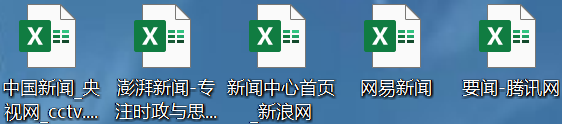
主要步骤：

（1）、利用爬虫工具“八爪鱼”对新闻网站上的信息进行爬取

（2）、通过python程序对导出的数据进行格式化，统一为**键-值对**形式的数据以便于后续分析

（3）、将整理过的数据集保存在文件中

数据来源有三部分：一部分来源于对对央视网新闻、澎湃新闻、新浪新闻、网易新闻和腾讯网五个新闻网站的数据爬取；一部分来源于提供的weibo\_dataset.zip；一部分来源于github平台上获取的CHEF文件夹，包含对微博辟谣平台的数据爬取，以下是内容展示



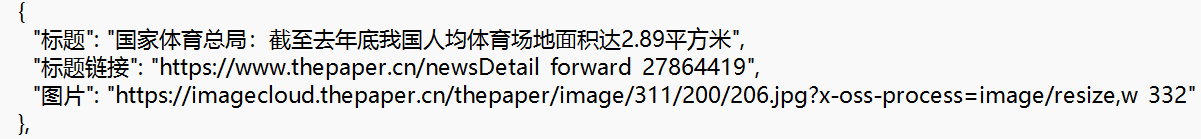




### 数据预处理

## 数据格式化:

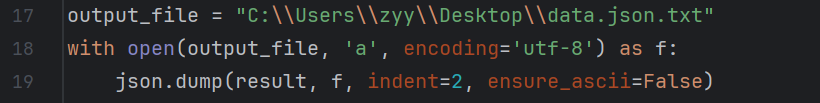
* 图片中数据的基本格式如下：



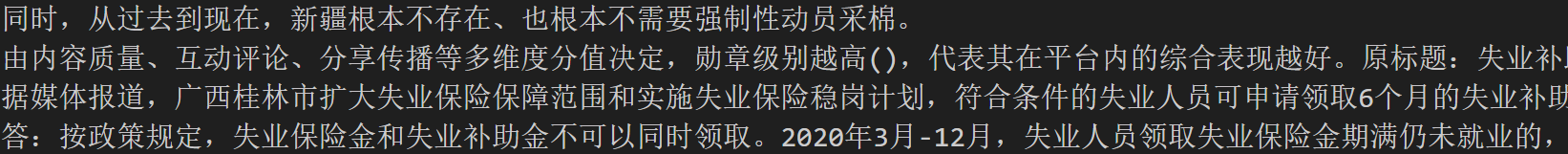
利用pandas库对excel表格进行读取并通过循环将数据以entry的形式存入列表



Index会遍历excel表格的每一列，而row则会获取相应列单元格中的值，用键值对将数据储存在一个entry中，然后再追加到result列表的末尾，对表格中的每一行都进行如上操作。



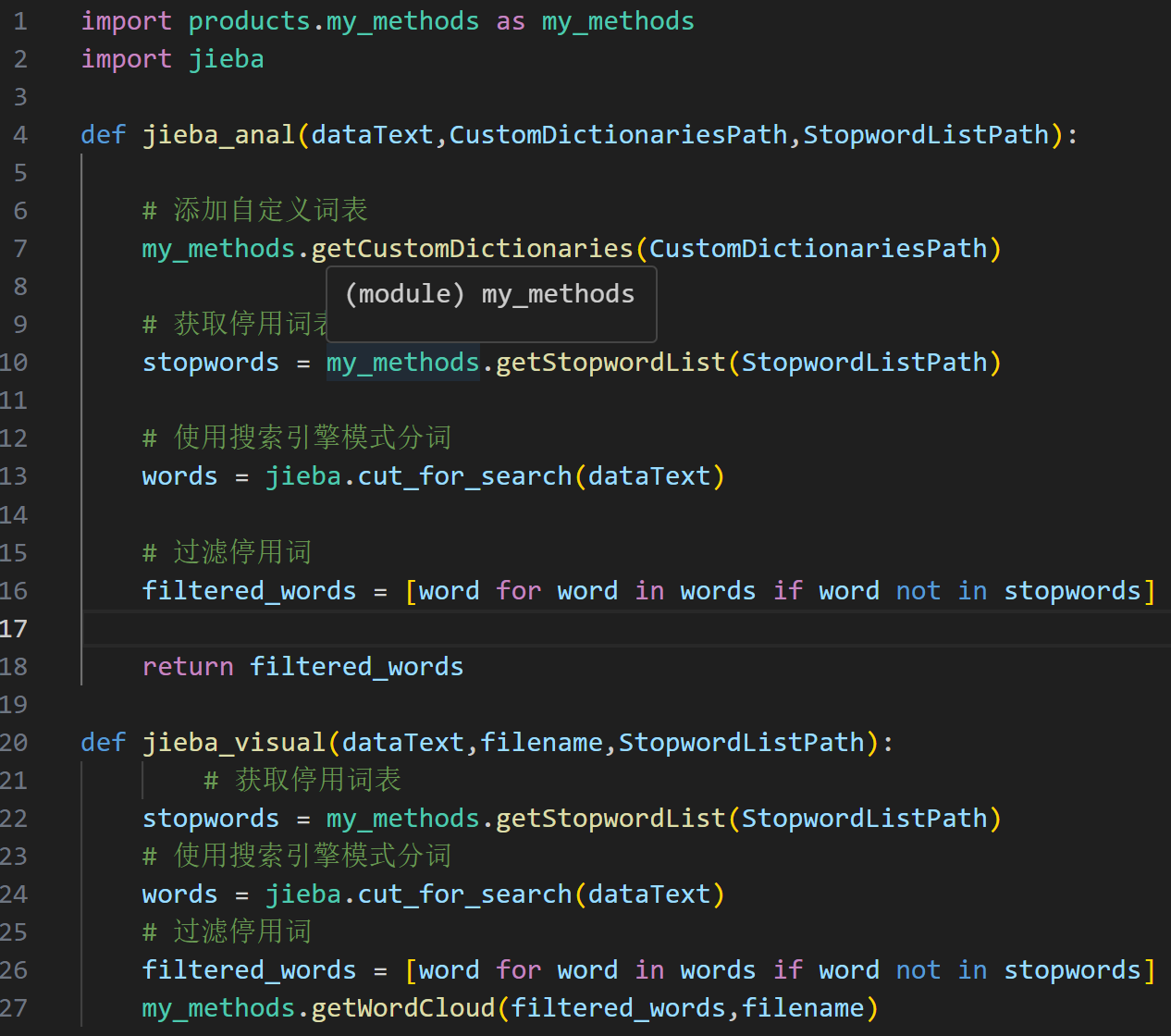
利用json.dump()将result中的数据保存至指定文件

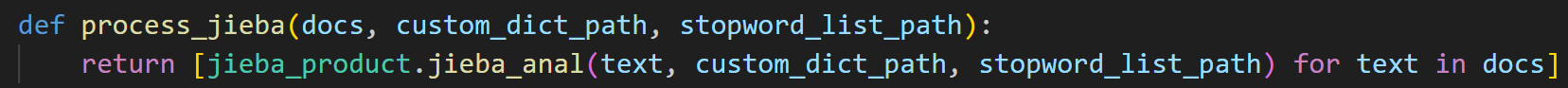
* 使用正则表达式等方法将如下图片
* 处理为:
* 这样的txt文件

## 文字预处理:

2.1使用jieba库进行处理:

* 使用jieba的搜索引擎模式进行分词,同时去除停用词

-

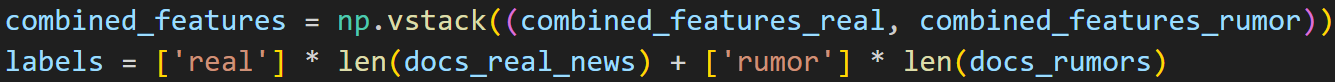
-

- 可视化以便检查分词与去停用词结果

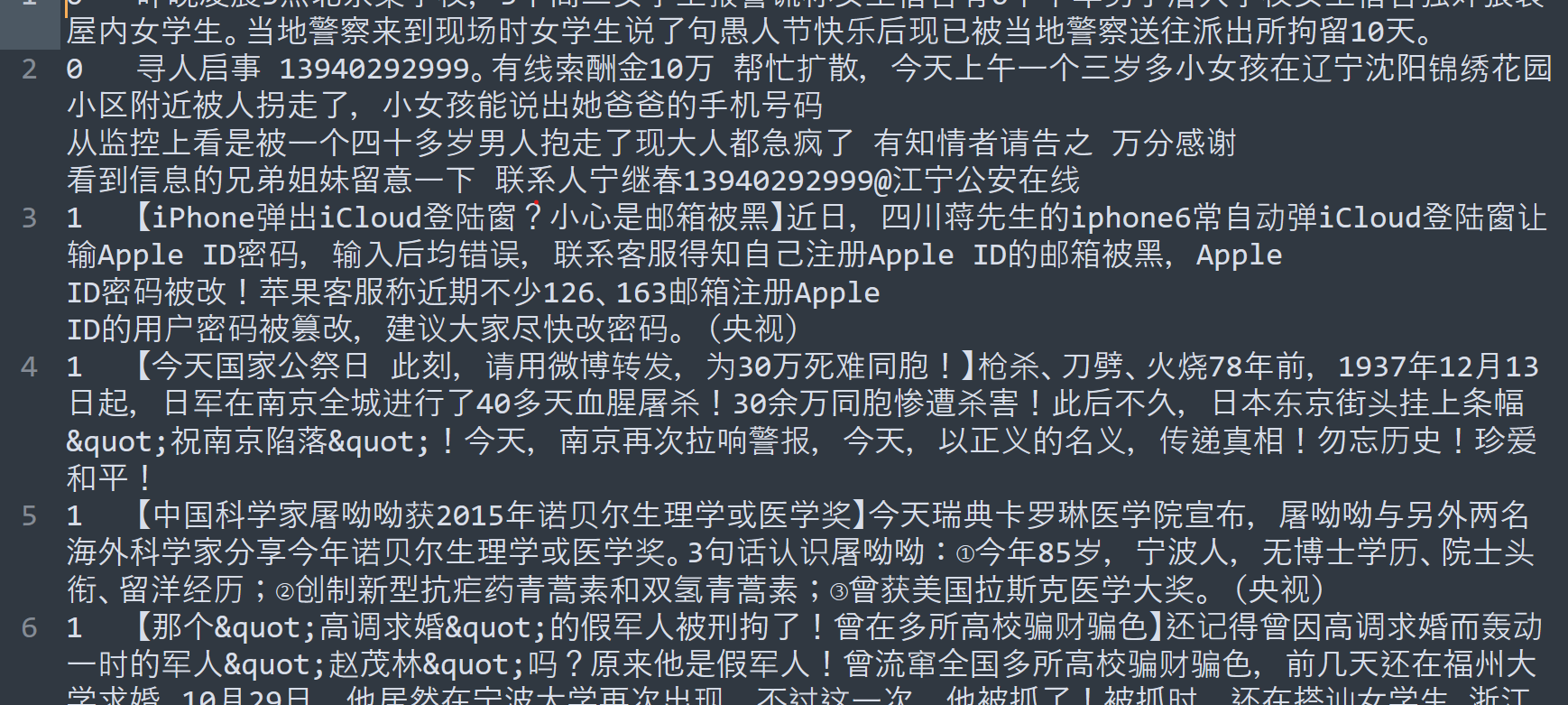
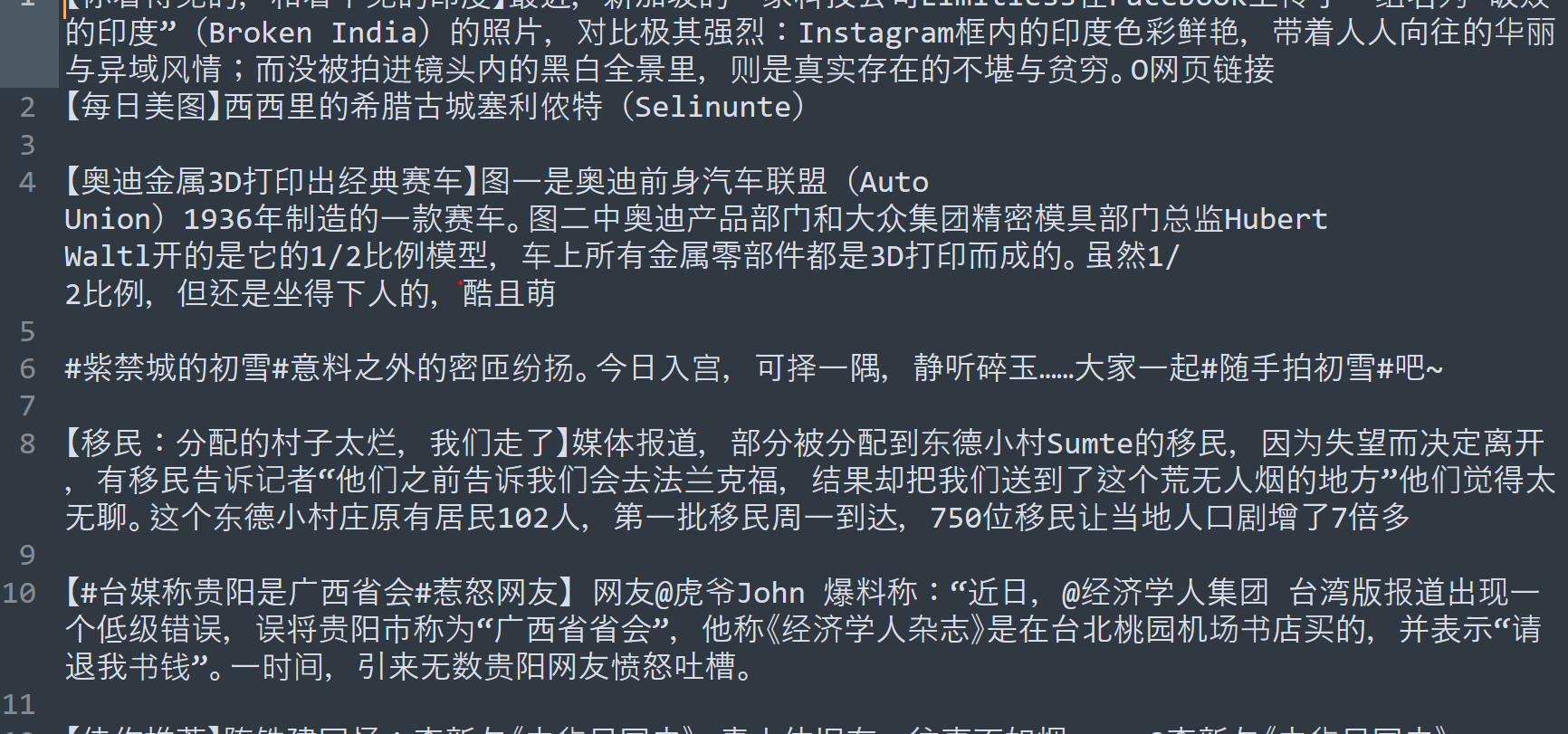
- 以下分别是真/假新闻的jieba可视化



2.2为真/假新闻设定标签:



2.3对tweets文件夹里的train\_nonrumor.txt和train\_rumor.txt两个文本做处理，只将所有的文本内容提取，并和它的lable（{0}为假，{1}为真）以乱序的形式存入all\_data.txt文本里。以下是部分数据展示：

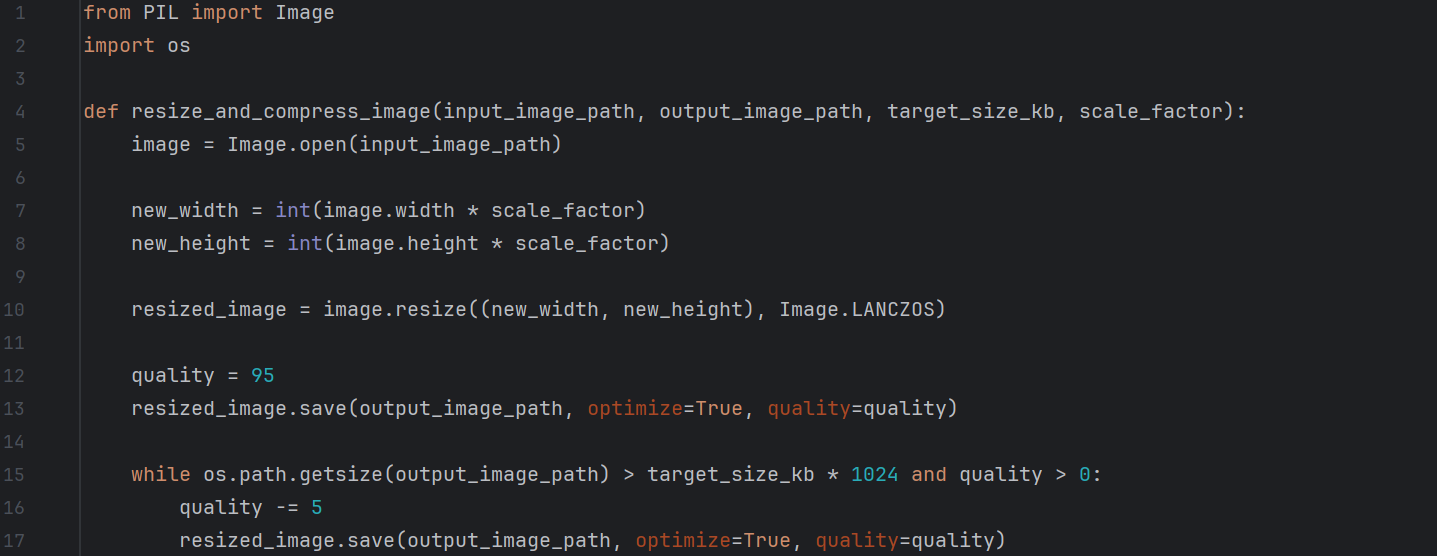


## 3.图片预处理:

为了使后续的图片特征提取，我们先将图片进行预处理，调整图片的大小和对比度。

* + 调整图片文件大小

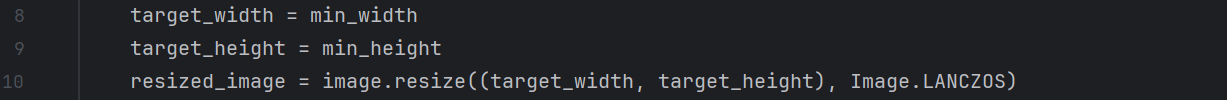
利用PIL库进行调整



设置目标大小target\_size\_kb，调用以上函数即可完成图片文件大小的调整。

* + 调整图片像素大小

还是用到PIL库进行处理



以缩小为例，设置目标的长宽，调用resize方法进行调整即可

* + 调整图片对比度

利用PIL可以轻松调整图片对比度

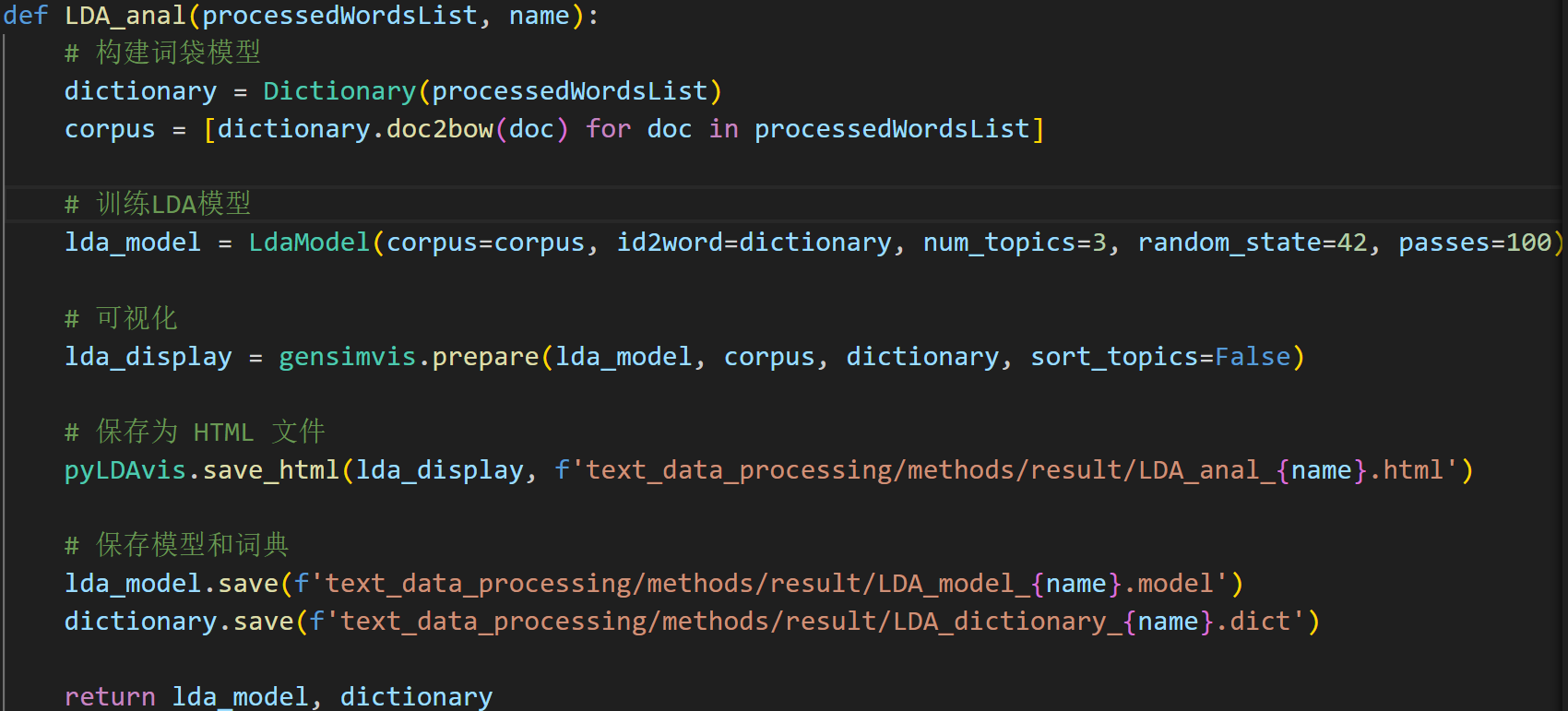
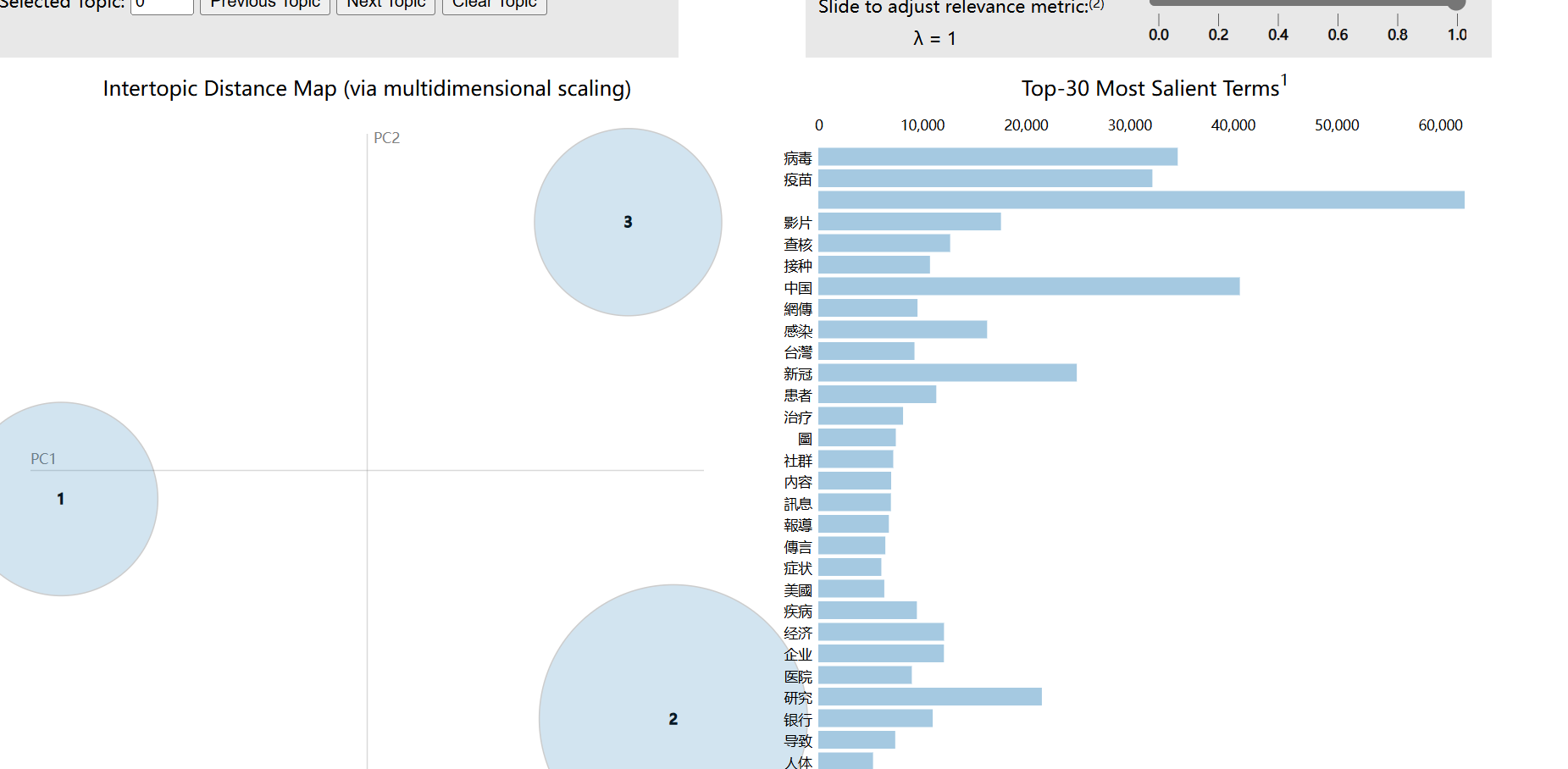


设置目标对比度直接调用相应方法即可

### 特征提取

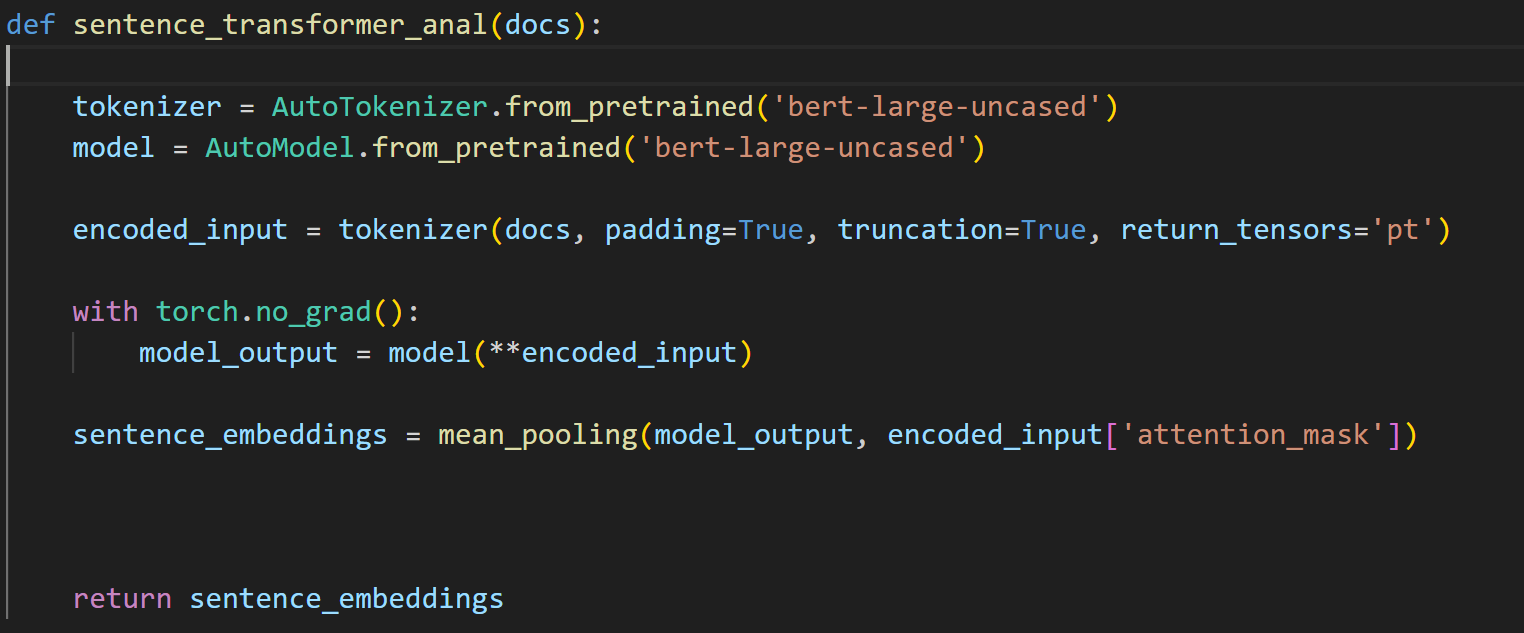
## 文字特征提取:

* 1. **使用LDA模型提取主题并进行可视化**

**** 

首先直观看出虚假新闻的主题内容和真实新闻的区别,重要的是得到lda模型和词典以供后续使用.

* 1. **使用 Hugging Face 上的 transformers 库和 PyTorch 来加载预训练的 BERT 模型，并生成给定文本的句子嵌入**



获得文档的向量表示以便后续使用

## 2, 图片特征提取:

主要工具：百度图片特征提取接口

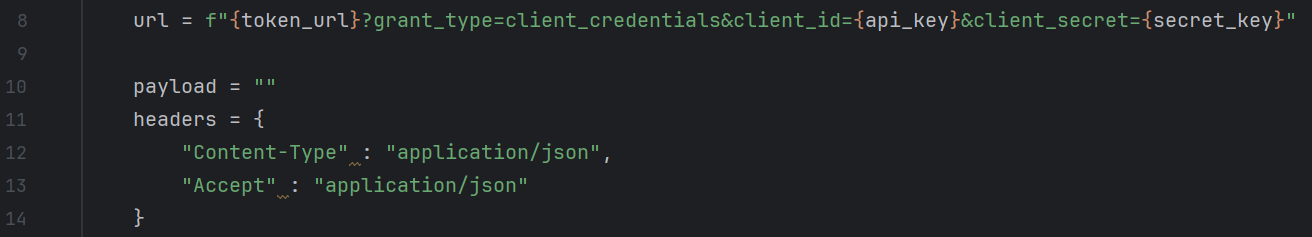
利用此前格式化后的数据可以得到新闻所对应的图片集，将图片的地址接入api相应位置进行循环处理，将分析结果截取需要字段内容写入文本文件。

**2.1 通行证获取**

在使用百度的接口之前要先获取接口的access\_token



在百度官方网站创建应用后，将api\_key、secret\_key和相应网址填入url并使用url进行access\_token的获取



添加payload和headers以便能够精确获取我们想要的接口

利用requests库进行获取



将response转为.txt后进行打印截取access\_token部分即可

**2.2 获取task\_token**

单张图片的特征分析需要用到单个任务的task\_token



请求response的时候有四个参数需要填入

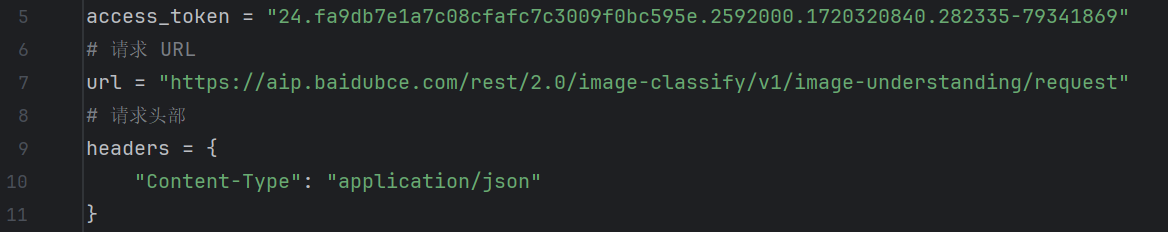
url：接口网址

headers：请求头部

params：access\_token

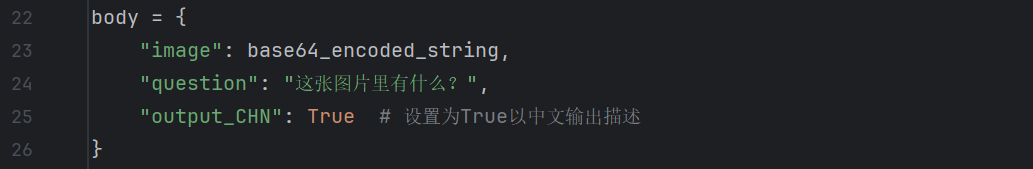
data：请求参数（body）

先记录url 、access\_token和headers

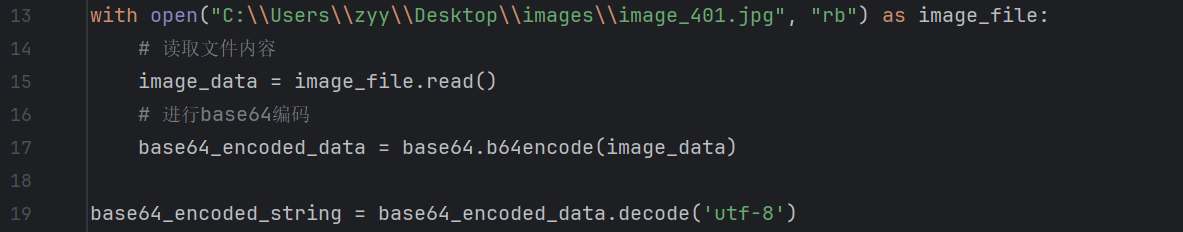


data的内容包括图片内容、需要得到回答的问题和输出形式（以中文输出）

用body{}对data进行书写



用base64库对图片内容进行base64编码，再进行utf-8解码



在得到response之后将它转为字典进行task\_token字段的提取即可

**2.3获取图片描述**

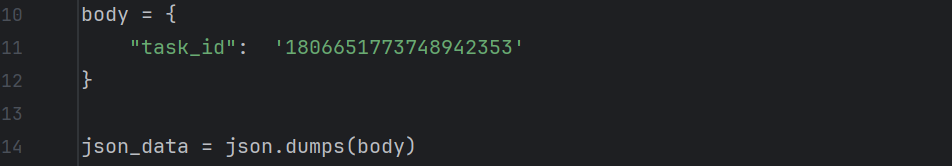
首先记录access\_token和接口网址url



这里得到response的方式和上一步骤类似，也是需要填入四个参数url、headers、params以及data

其中headers和data有所变化



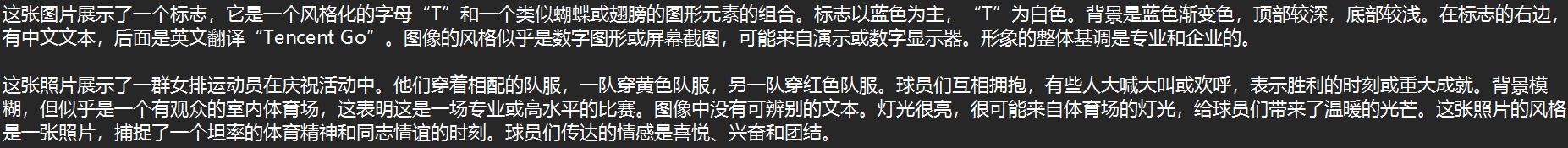


然后发送POST请求即可

2.4**最后将结果截取需要的description部分写入文件中**



如下:

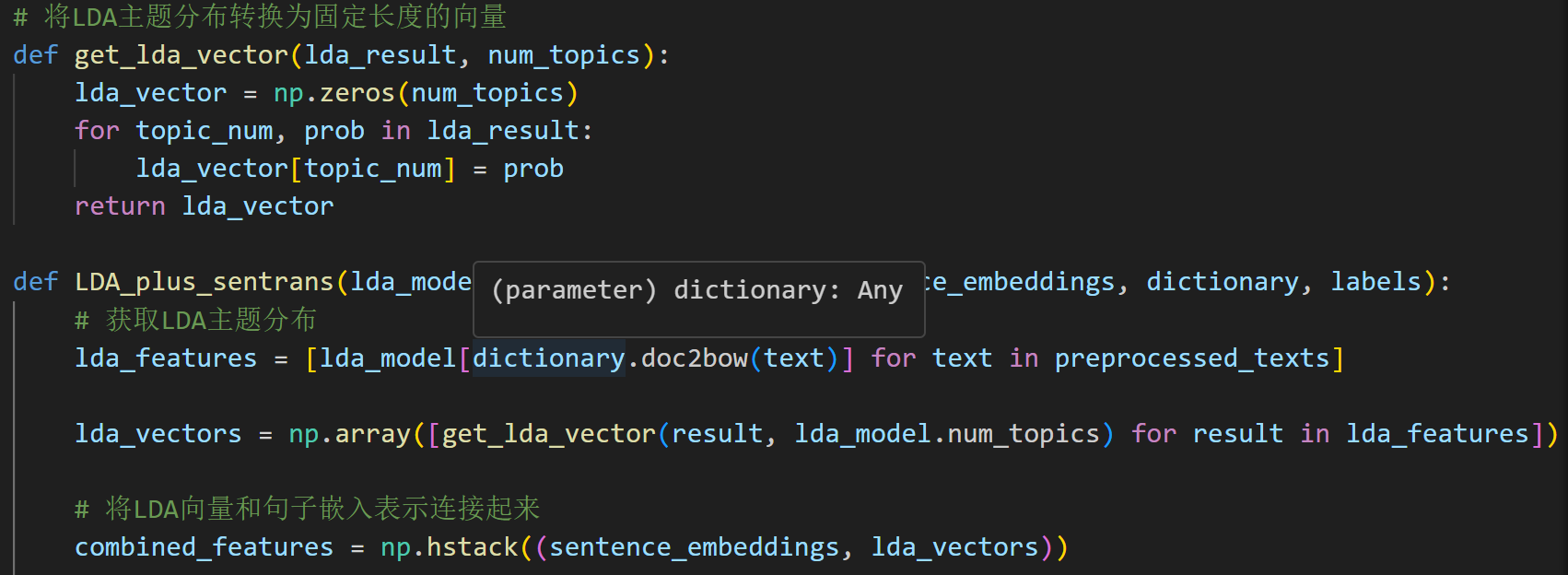


### 数据分析

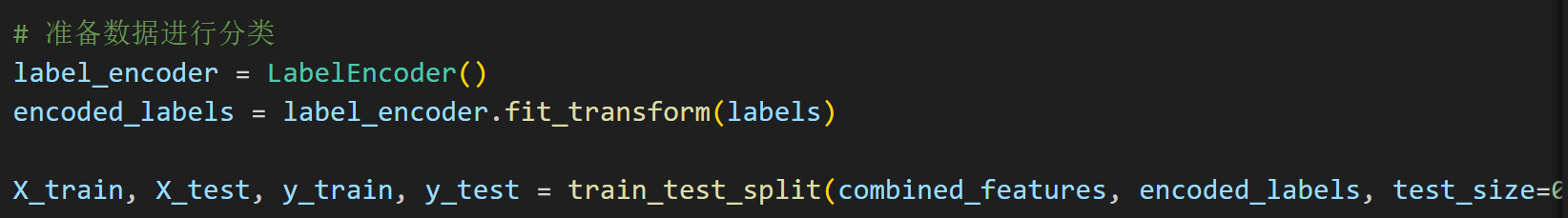
#### 以下将从两个方法入手:

## 1.将LDA与得到的句子向量结合训练模型

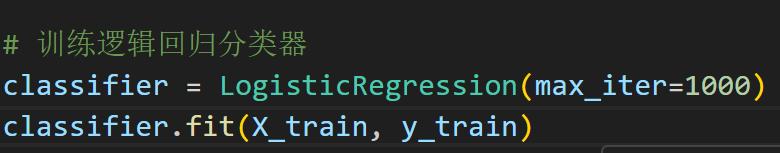
- **将LDA的主题分布的向量与句子嵌入的向量连接起来:**



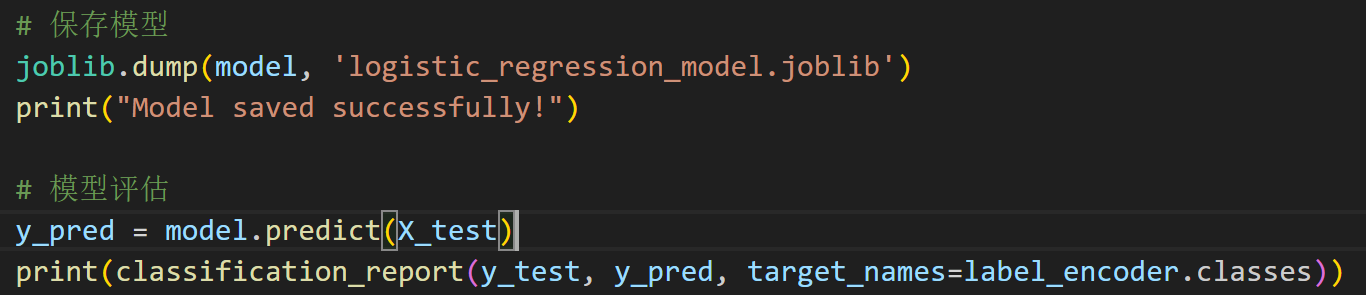
* **进行数据分类:**



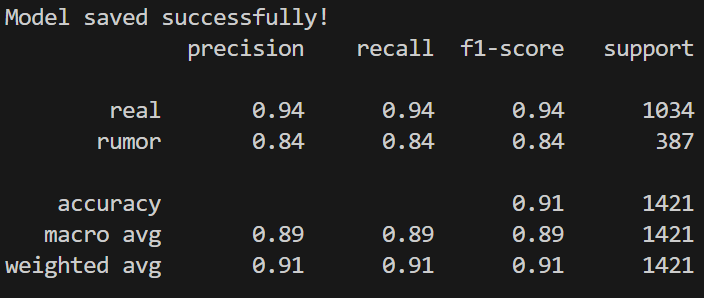
* **训练逻辑回归分类器:**



* **最后保存模型并进行模型评估:**

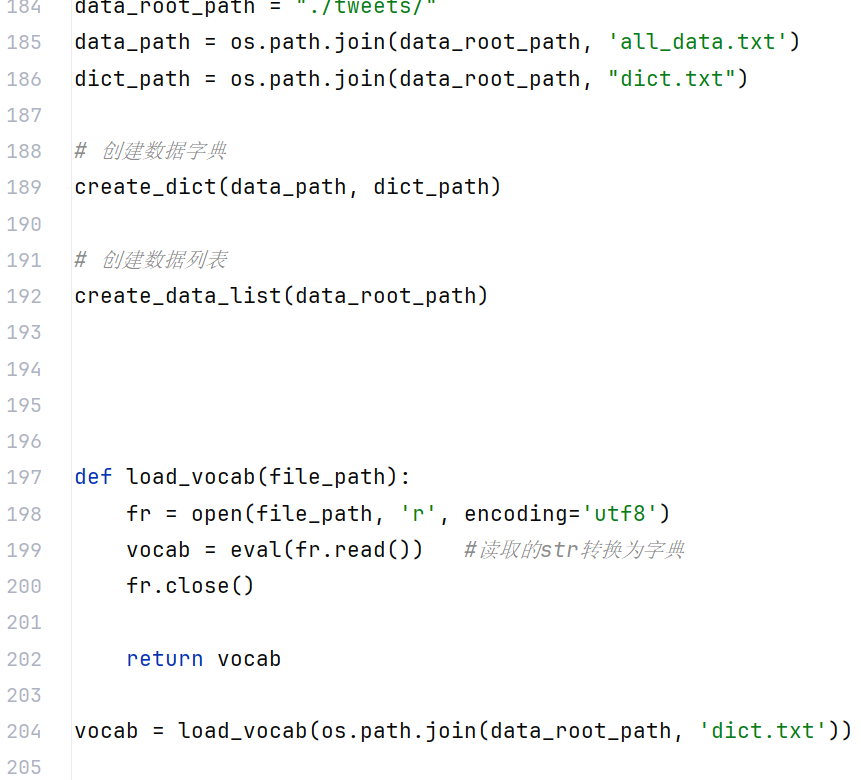


得到结果如下：

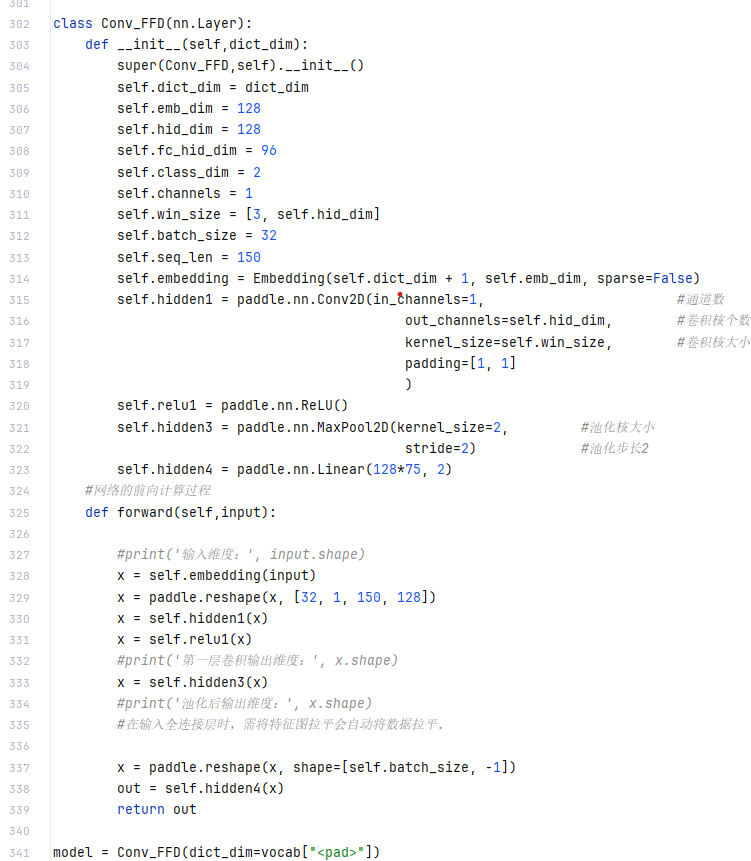


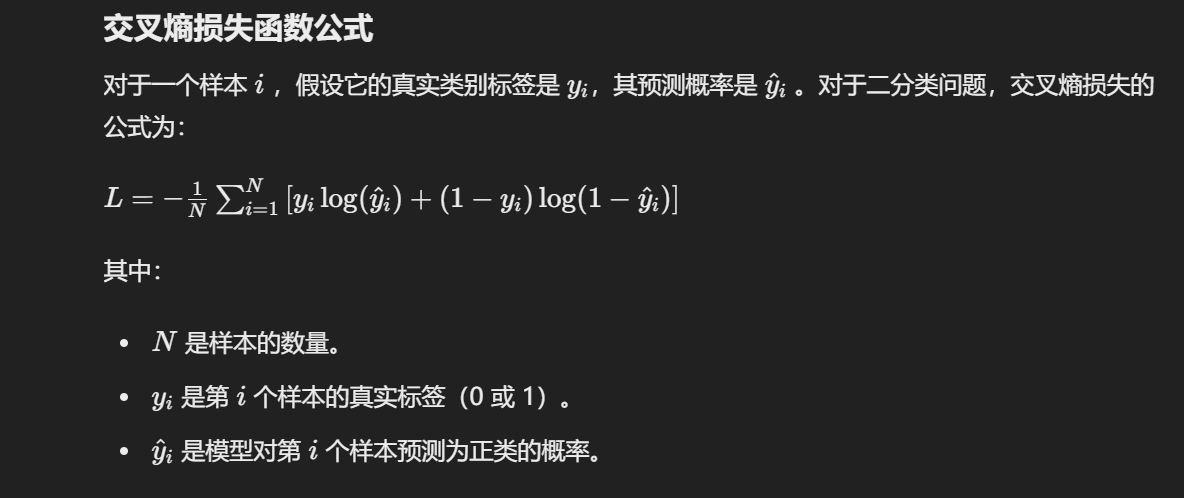
## （Conv-FFD）

Conv-FFD将中文短文本中的每个字直接作为基本处理单元。使用特征提取滤波器Conv扫描句子中的每个单元字符，通过卷积运算从一个字符和相邻字符中提取联合特征表征。最后，通过全连接层映射最终获得检测结果。为了快速将字符级特征表示合并为句子级特征表示，选择了CNN。通过制作数据字典，对文本进行处理：



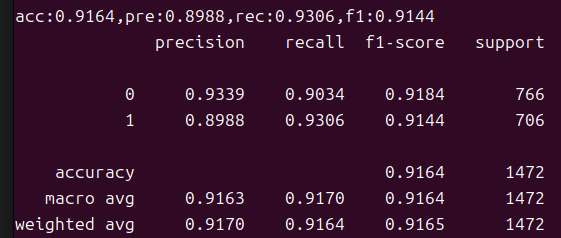
# 通过对A Deep Learning-based Fast Fake News Detection Model for Cyber-Physical Social Services论文的浮现，利用基于卷积的神经计算结构实现了快速假新闻检测（Conv-FFD），部分代码实现如下：



将train\_data用于模型训练，分别计算模型的准确度(acc)和**信息熵损失(loss)**,在经过近十分钟的运行后，得到如下结果：



再对召回率（recall），精确率（precision），F1值（f1-score)进行计算，结果如下：



将test\_data用于检验，代码实现如下：

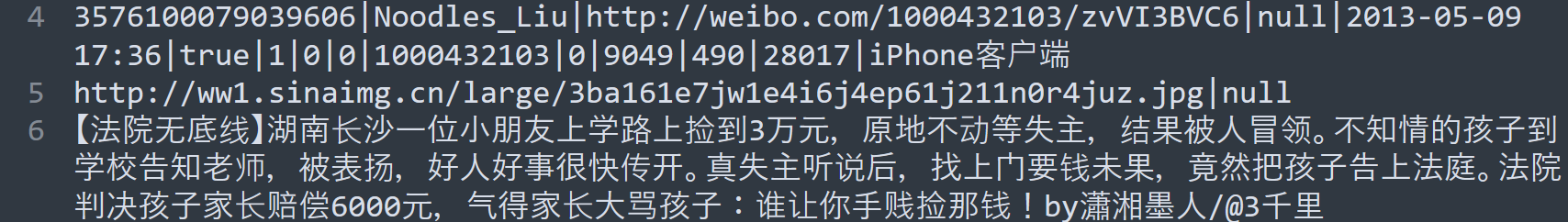


最后，该检测文本谣言的模型训练完成。

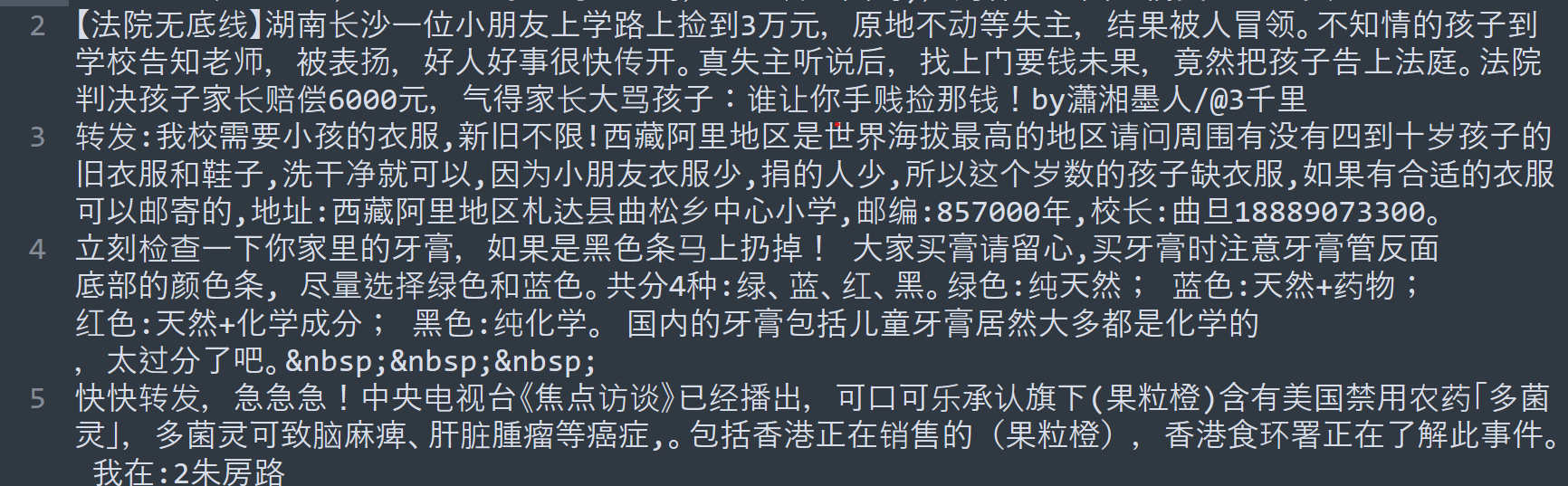
### 图文融合

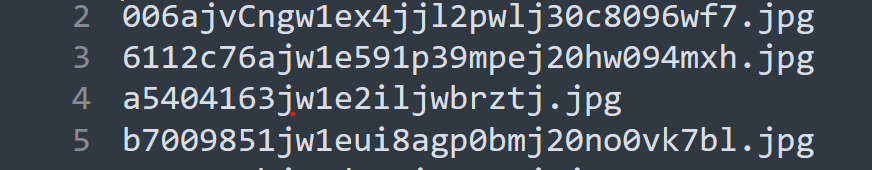
1. 数据预处理

使用python将原本格式为 {信息来源 \n 图片 \n 文本信息}

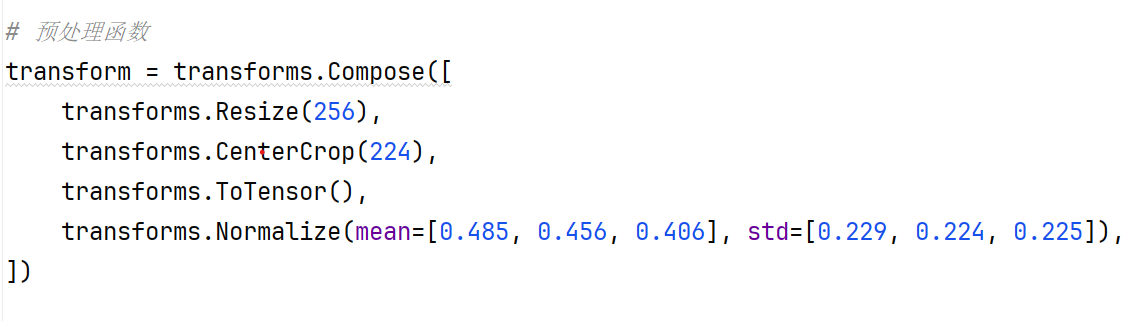


转化为一个文本对应一个图片，其中有多个图片对应一个文本的数据以及未保存在文件夹里的图片地址进行了删减，只保存了第一个可以在文件夹中找到的图片。文本的序号可以和图片的序号进行对应。以下是部分数据的结果：





为了保证图片格式的兼容，对图片格式进行预处理：

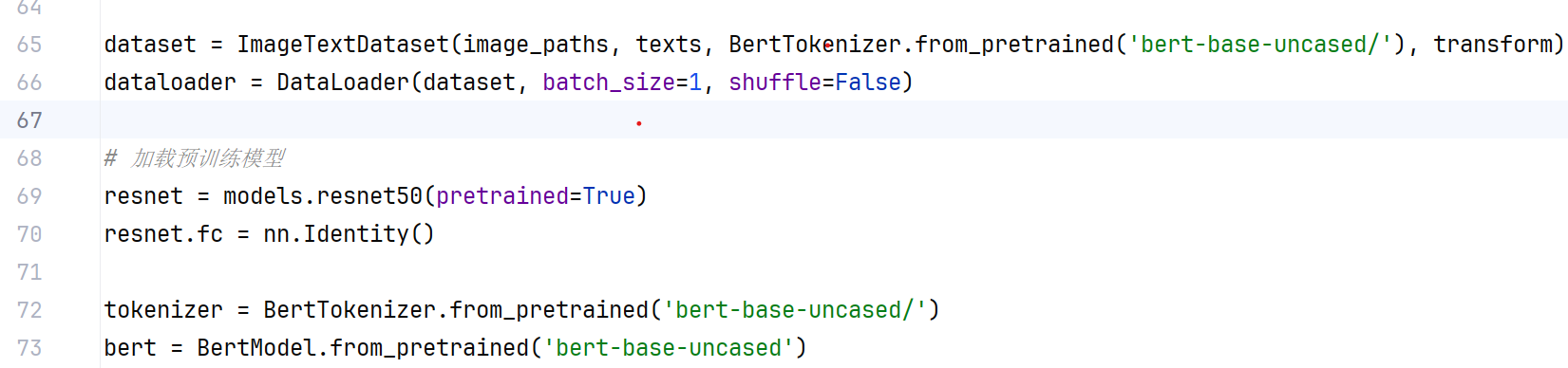


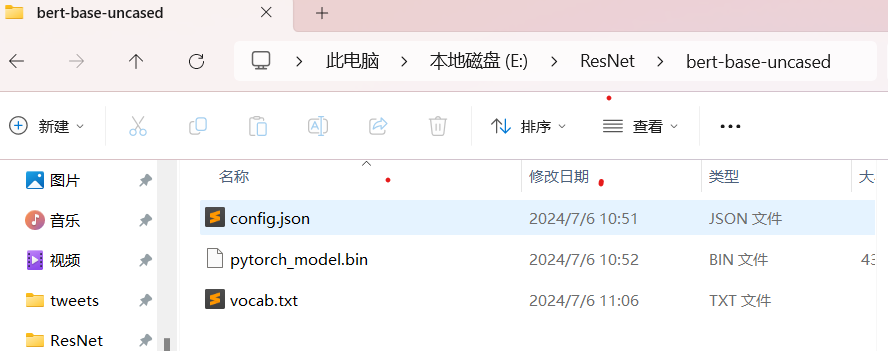
1. 调用模型

将所有数据的文本和所有图片地址的文本传入模型中，以下部分涉及到两个模型，现做相关原理解释：

ResNet-50是一个卷积神经网络模型，通常用于图像分类任务。在图文相似度检验中，可以使用ResNet-50模型提取图像的特征表示。BERT是基于Transformer模型架构的预训练语言模型，对于图文相似度检验，可以使用BERT模型将文本序列编码为向量表示，综合使用ResNet-50提取图像特征和BERT提取文本特征，可以通过将图像特征和文本特征投影到共同的空间，然后计算它们之间的相似性来进行图文相似度检验。通过比较这些特征向量之间的相似性度量，可以评估图像和文本之间的相关性和相似性。

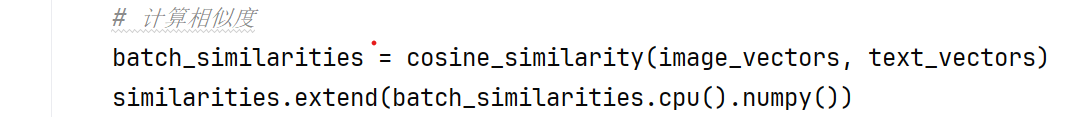
以下是需要的相关文件以及相关代码：





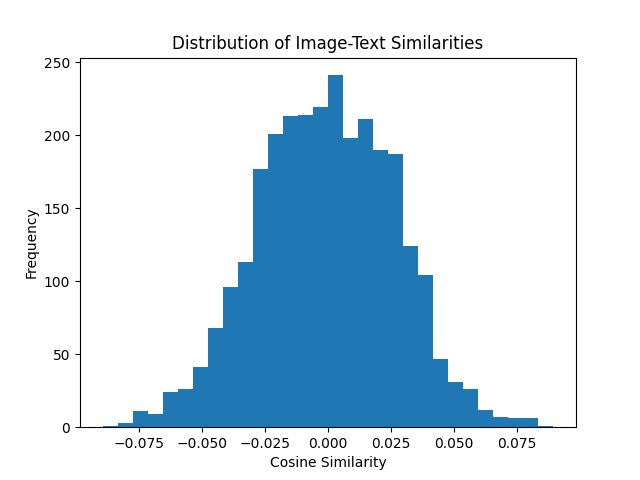
1. 得到图文相似度

通过获取文本向量和图片向量后，并进行对齐，进入循环计算相似度



将最终结果存入csv文件里。

1. 绘制分布图



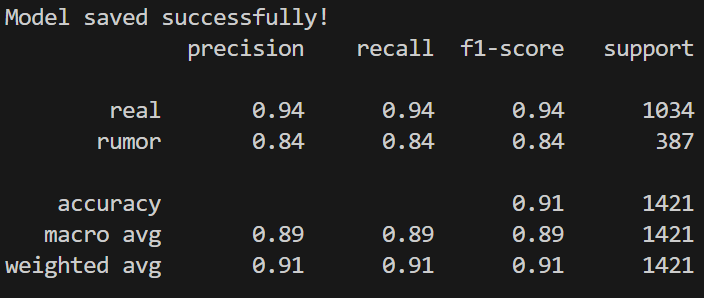
上图为非谣言的图文相似度分布



上图为谣言的图文相似度分布

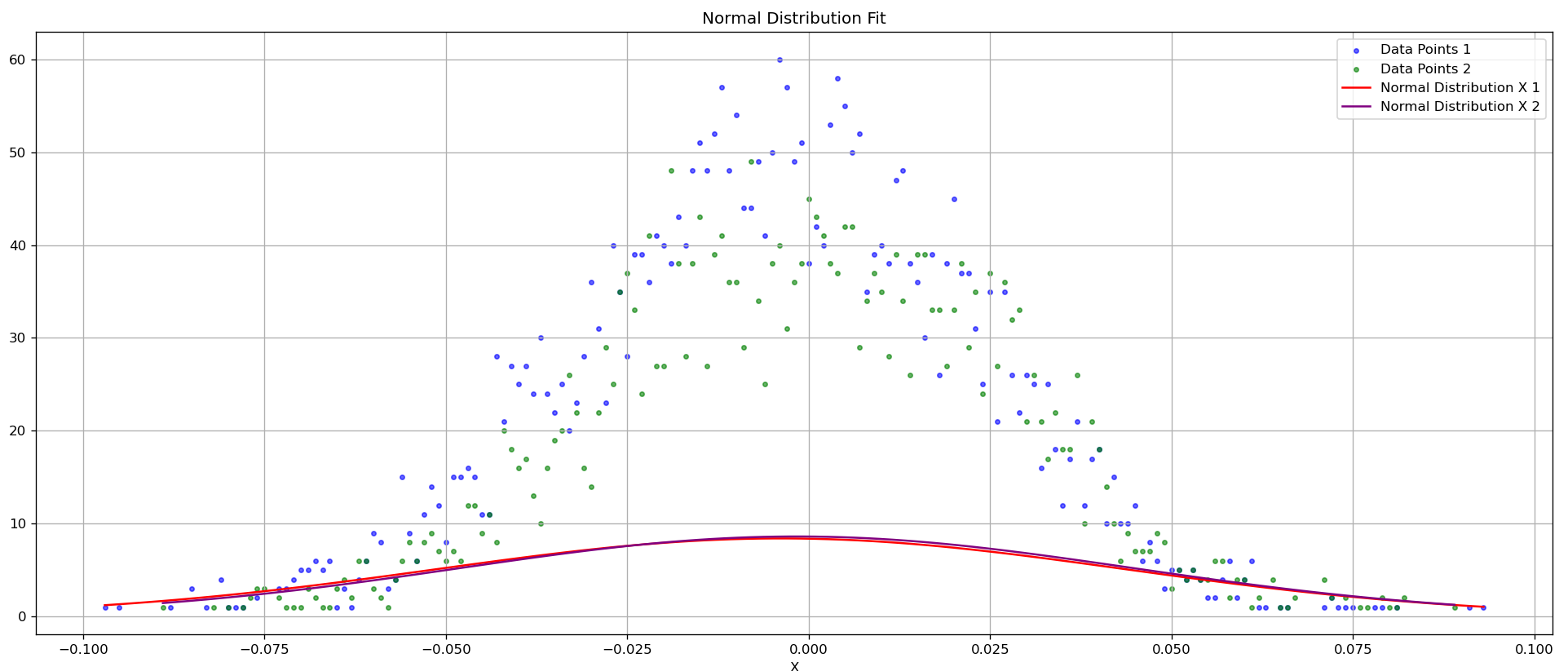
### 结果呈现与报告编写

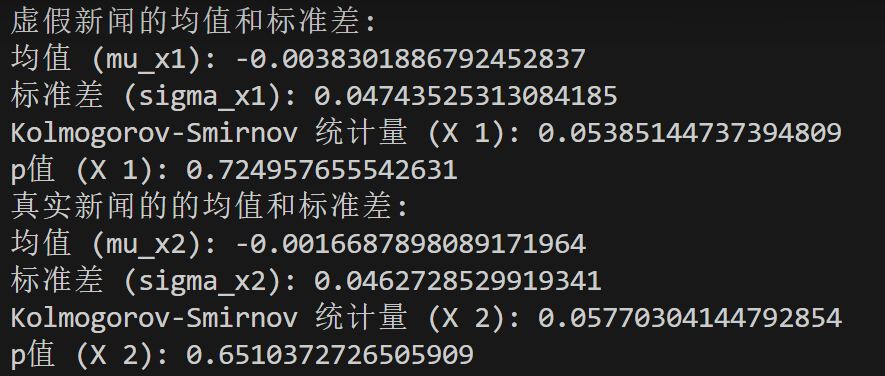
* 首先得到文字分析模型的准确度及一些参数:



设该准确度为**P(acc)**

* 其次获取图文相似度的分布并使用程序近似为正态分布并计算出均值与标准差并进行拟合优度估计得到p值如下:





p值均大于0.05，表示我们不能拒绝数据来自正态分布的假设。因此，可以认为两组数据符合正态分布的假设。

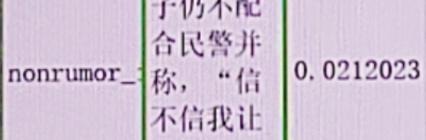
我们通过大数定律以该大样本的分布近似于总体分布.

设在x = X处的概率为P(X)

* 处理过程:

给定一条新闻,先使用文字分析模型判定其真假性,并得到**P(acc),**然后得到其图文相似度,从而由总体分布得到**P(X),** 由公式**P = (3 \* P(acc)) / 4 + P(X) / 4** 得到该次判定的准确率.

* **例:**



文字模型判定: 谣言

实际: 非谣言



可知**P(acc)** = 0.9164

由正态分布知 **P(X)** = 0.0023

可计算**P** = 0.688

可知图文融合在一定程度上具有修正效果

### 潜在问题

* 图片的特征提取与OCR的利用不佳.
* 将LDA与得到的句子向量结合训练模型的过程虽具创新性但效率不高.
* 最终准确率的计算公式中的参数有待改进.