周工作总结20190506-20190510

本周主要的工作是开发simhash离线验证的相关代码，主要是用到了es。同时验证了上周训练的fasttext模型的分类效果。

技术积累

1. requests发送请求

   一种是以form表单形式发送post请求，只需要将请求的参数构造成一个字典，然后传给requests.post()的data参数即可。

```
# 获取小米有品的分类信息
headers = {
        "Host": "youpin.mi.com",
        "Content-Type": "application/x-www-form-urlencoded",
        "Referer": "https://youpin.mi.com/",  # 必须带这个参数，不然会报错
        "User-Agent": "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/66.0.3359.181 Safari/537.36",
            }
url = "https://youpin.mi.com/app/shopv3/pipe"
form_data = {"data": '{"result": {"model": "Homepage", "action": "BuildClass", "parameters": {"id": -6}}}'}
results = requests.post(url, data=form_data, headers=headers).text
print(results)
```

   二种是以json形式发送post请求：可以将一json串传给requests.post()的data参数

```
# http://jinbao.pinduoduo.com/index?page=1里面的分类,
import requests
import json
headers = {
    "Content-Type": "application/json; charset=UTF-8",
    "Referer": "http://jinbao.pinduoduo.com/index?page=5",
    "User-Agent": "Mozilla/5.0 (Windows NT 6.1; WOW64)
AppleWebKit/537.36 (KHTML, like Gecko) Chrome/67.0.3396.87
Safari/537.36",
    }
url = "http://jinbao.pinduoduo.com/network/api/common/goodsList"
pyload = {"keyword": "", "sortType": 0, "withCoupon": 0,
"categoryId": 16, "pageNumber": 1, "pageSize": 60}
response = requests.post(url, data=json.dumps(pyload),
headers=headers).text
print(response)
```

2. eval(expression[, globals[, locals]])，eval同时可以把字典的字符串转成字典类型

   eval() 函数用来执行一个字符串表达式，并返回表达式的值。

   expression是表达式，global是变量作用域，全局命名空间，如果被提供，则必须是一个字典对象，locals是变量作用域，局部命名空间，如果被提供，可以是任何映射对象。如x=7,eval('3*x')则可以返回21。

3. ES使用笔记

   - query：query字段要写的是你要查询的内容，也就是某个字段的值。
   - qfs：qfs字段放的是你要查询的字段，跟query是配合使用的。比如

     ```
     {
         "query":"6513479713144705174 14581765041235735119",
         "qfs":"simhash_slide",
         "size":10,
         "getRelativeScore":true,
         "queryMinimumShouldMatch”:”2"
     }
     ```

   - queryMinimumShouldMatch: 这个字段的值分两种，整数和百分比。整数指的是匹配到的个数，比如例子中的”simhash_slide“是以空格分词器分割的，长度为2，queryMinimumShouldMatch也是2，也就是要每个数字都一样的那条记录。如果是百分比的话， 指定可以返回的搜索结果最小匹配度，大于等于此匹配度的搜索结果才可以返回。

4. 二进制和十进制互转

   二进制转十进制：

```
simhash_overall =
'00101011110111011101100000001101001001100110111000100001100110
0'
simhash_all = int(str(simhash_overall), 2)
```

   十进制转二进制的时候，发现转换完以后的二进制有个'0b'开头，比如

```
binary =
'0b10101111011101110110000000110100100110011011100010000110011010
0'
str(int(binary, 2))
```

异常问题

1. the JSON object must be str, not 'bytes'。

   刚遇到的时候以为是python2和python3的区别，最后发现是python3的版本不一致，用的那个项目3.7和3.5都不行，只有python3.6能正常执行，还没有发现问题。

2. mac上执行，pip install fasttext，报错No module named Cython.Build。

   解决方法：1. pip install --upgrade cython

   然后又报错：

```
clang: warning: libstdc++ is deprecated; move to libc++ with a
minimum deployment target of OS X 10.9 [-Wdeprecated]
    ld: library not found for -lstdc++
    clang: error: linker command failed with exit code 1 (use -v
to see invocation)
    error: command 'g++' failed with exit status 1
```

   看一篇博客说是g++版本不够，所以准备升级一下试试。

```
brew update
 brew reinstall gcc # 安装最新的gcc8.3.0.2
安装完是：/usr/local/Cellar/gcc/8.3.0_2: 1,414 files, 288.5MB
```

   用这个命令继续装：env CC=/usr/local/Cellar/gcc/8.3.0_2/bin/gcc-8 pip install fasttext

结果还是报错：

gcc-8: error: unrecognized command line option '-stdlib=libc++'

error: command '/usr/local/Cellar/gcc/8.3.0_2/bin/gcc-8' failed with exit status 1

最后找到了一个解决方法就是在~/.bash_profile文件里加上几行：

```
alias gcc='/usr/local/Cellar/gcc/8.3.0_2/bin/gcc-8'
alias g++='/usr/local/Cellar/gcc/8.3.0_2/bin/g++-8'
alias c++='/usr/local/Cellar/gcc/8.3.0_2/bin/c++-8'
```

这里有个问题是这个gcc的路径要写清楚，要不然会报找不到gcc-8.

不过fasttext还是没装好，后面有时间再继续研究吧。

个人反思

1. 遇到的问题很多，但是很多都是没有深究到最后，所以后面应该有一个专门的**未解决问题列表**，用一个专门的文档或者github的文件夹来维护，追踪最后的结果。
2. 本周在开发simhash离线效果验证的时候，要往es里灌数据，计算了下每条要0.35s时间，感觉还不错啊，挺快的，但是一般的时间消耗其实只有百分之一或者千分之一，一方面是缺乏经验，一方面是对代码没有精益求精，所以这里要提醒下自己，**代码的效率和效果同样重要**，后面要在关注质量的同时，也要多关注效率。