信息检索实验四 VSM & BM25

实验硬件 & 软件

i5-6200U + python3.7 + bs4

实验内容

本次实验要求在实验3的基础上对查询结果进行相关度打分排序。实现VSM & BM25.

实验步骤

- 首先看了看能不能复用 hw3 的代码，后来放弃了
- 然后看了看需不需要 nlp 之类的 tokenization，好像也不太需要 正则就够了
- 然后，给的查询文本是 query171-225.txt 是xml的，转换成json并转小写 trans_query.py -> query171-225_cleaned.txt
- get_solve.py

读取tweets，建立posting_list，两个相关函数计算

```
def VSM_F(c_w_q,c_w_d,tweet_l,df_w,b=0.5,avdl=20,M=len(tweets)):
    return c_w_q*(log(1+log(1+c_w_d))/(1-b+b*(tweet_l/avdl)))*(log((M+1)/df_w))
def BM25(c_w_q,c_w_d,tweet_l,df_w,k=8,b=0.5,avdl=20,M=len(tweets)):
    return c_w_q*((k+1)*c_w_d/(c_w_d+k*(1-b+b*(tweet_l/avdl))))*(log((M+1)/df_w))
```

因为同一个查询与同一篇文章的 `f(q,d)` 是多个单词的累加，所以上述函数是：

$$c(w,q)\frac{\ln[1+\ln[1+c(w,d)]]}{1-b+b\frac{|d|}{avdl}}\log\frac{M+1}{df(w)}$$

$$c(w,q)\frac{(k+1)c(w,d)}{c(w,d)+k(1-b+b\frac{|d|}{avdl})}\log\frac{M+1}{df(w)}$$

可能会遇到，查询的是单词的一部分，这时候有两种情况：

一、用户所输入的单词是目标单词的一部分，且我们的posting—list没有用户输入的单词

二、用户所输入的单词是目标单词的一部分，但是我们的posting—list有这个单词

我们假设用户都是"聪明"人，不会有类似于 输入git ，找github 的情况

只是当用户输入的word不在posting—list中时才查找所有的posting—list

avdl = 108

两个方法的评估结果分别存在：

```
VSM_envalue_result.txt
BM25_envalue_result.txt
```

贴一下BM25的成绩：

```
tf
    avdl = 20
    MAP = 0.5330725049635453
    NDCG = 0.7065456019418237

    avdl = 108
    MAP = 0.5226944141019757
    NDCG = 0.6948554310373035
df
    avdl = 108
    MAP = 0.5532475063771154
    NDCG = 0.7296616730491001
```

VSM的：

```
tf
    avdl = 20
    MAP = 0.5342015345607777
    NDCG = 0.7038275670022469
df
    avdl = 108
    MAP = 0.5608640193310123
    NDCG = 0.7303678927365987
```

PS

那个，第一版算的是tf。。。。已更新。