CA4009 Search Technologies Lab #4 & 5

Date: 02/12/2019

Name(s): Maksims Kompanijecs - 15306971, Matthew Farelly - 15366246

4.Experimenting with Query Expansion in IR

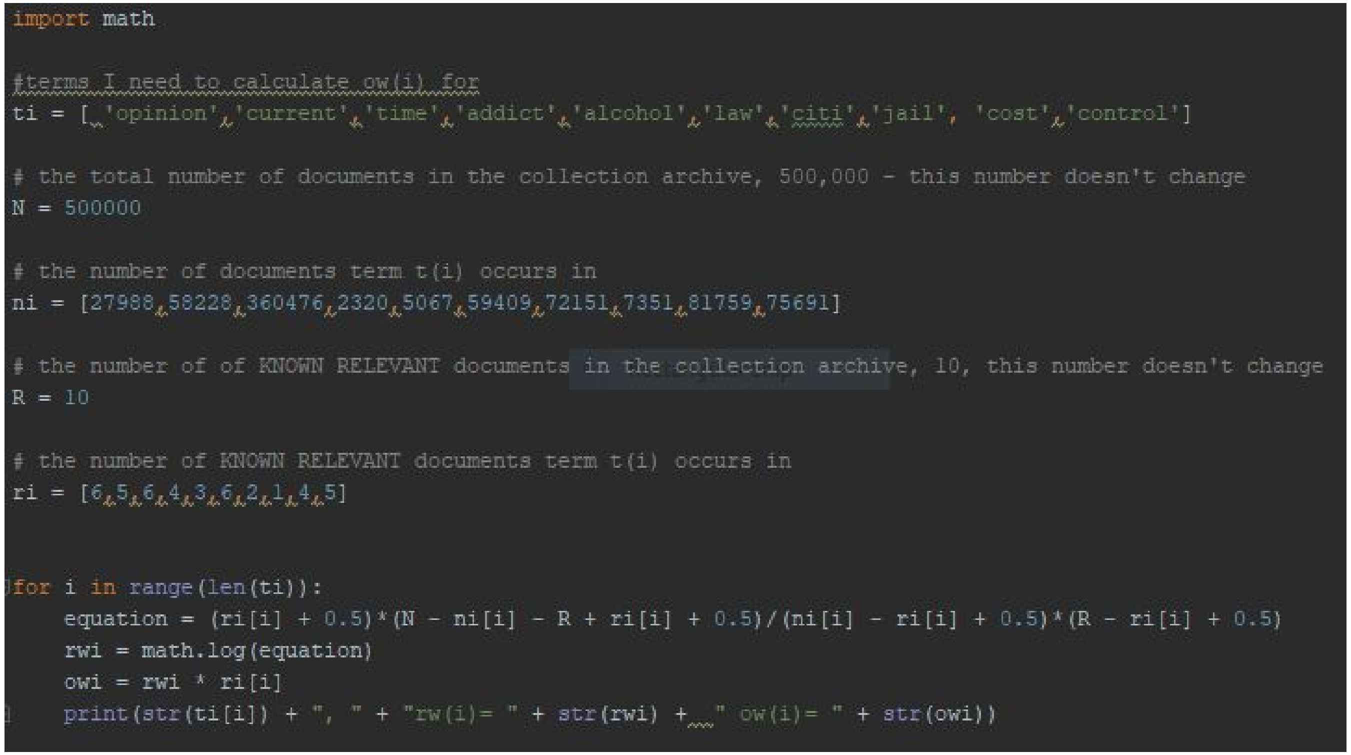
Query topic from TREC collection selected: 360 - drug legalization benefits
Using BM25 ranking functions weighing terms were **b = 1.2** and **k = 0.75**

The documents retrieved were as follows:

	Article	Description
1	**LA031289-0044**	article from March 12, 1989 about drug epidemic, stating that is time to legalize drugs and sell them in government controlled stores
2	**LA032590-0032**	article from March 25, 1990 about drug decriminalization in America and how it is the worst from of drug control
3	**FBIS4-57566**	article from 14 May 1994 on how Law 1008 has achieved achieved good results and met its goal of struggling against drug trafficking
4	**FBIS4-67248**	article from 14 May 1994 about ATIN AMERICA BOLIVIA Soliz Agrees With U.S. Ambassador on Drug Law
5	**FBIS3-9970**	article from 28 Feb 1994 detailing Interview with Colombian Prosecutor General Gustavo de Greiff about progress was toward the surrender of Cali Cartel members
6	**FBIS3-21631**	article from 28 February 1994 about surrender of Cali Cartel members and details in which prison where they will stay
7	**LA112089-0024**	article from November 20, 1989 detailing forms of controlled legalization of drugs by George P. Shultz,
8	**LA031990-0020**	article from March 19, 1990 about drug dealing and how it offers powerful economic incentives for the young
9	**FT931-16994**	article from 04 JAN 93 about bell tolls for drugs harmony
10	**FR940112-2-00078**	article from N/A about how claimant applied for disability benefits under the supplemental security income program

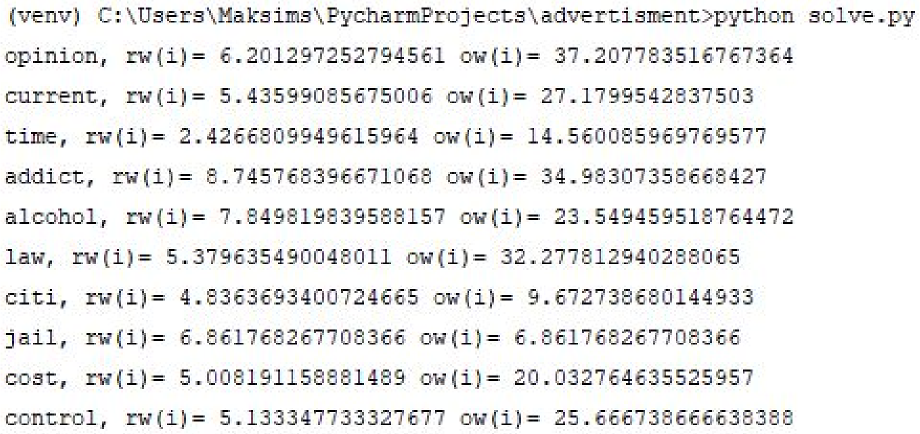
Calculating ow(i) values for terms that appear in the known relevant documents

I wrote a Python script that would calculate ow(i) and rw(i) value based on the most significant terms that I have selected. This saved me a lot of time having to calculate ow(i) and rw(i) values manually.



```
import math

#terms I need to calculate ow(i) for
ti = ['opinion','current','time','addict','alcohol','law','citi','jail', 'cost','control']

# the total number of documents in the collection archive, 500,000 - this number doesn't change
N = 500000

# the number of documents term t(i) occurs in
ni = [27988,58228,360476,2320,5067,59409,72151,7351,81759,75691]

# the number of of KNOWN RELEVANT documents in the collection archive, 10, this number doesn't change
R = 10

# the number of KNOWN RELEVANT documents term t(i) occurs in
ri = [6,5,6,4,3,6,2,1,4,5]


for i in range(len(ti)):
    equation = (ri[i] + 0.5)*(N - ni[i] - R + ri[i] + 0.5)/(ni[i] - ri[i] + 0.5)*(R - ri[i] + 0.5)
    rwi = math.log(equation)
    owi = rwi * ri[i]
    print(str(ti[i]) + ", " + "rw(i)= " + str(rwi) + " ow(i)= " + str(owi))
```

These are the rw(i) and ow(i) values generated from the script above.



```
(venv) C:\Users\Maksims\PycharmProjects\advertisment>python solve.py
opinion, rw(i)= 6.201297252794561 ow(i)= 37.207783516767364
current, rw(i)= 5.43599085675006 ow(i)= 27.1799542837503
time, rw(i)= 2.4266809949615964 ow(i)= 14.560085969769577
addict, rw(i)= 8.745768396671068 ow(i)= 34.98307358668427
alcohol, rw(i)= 7.849819839588157 ow(i)= 23.549459518764472
law, rw(i)= 5.379635490048011 ow(i)= 32.277812940288065
citi, rw(i)= 4.8363693400724665 ow(i)= 9.672738680144933
jail, rw(i)= 6.861768267708366 ow(i)= 6.861768267708366
cost, rw(i)= 5.008191158881489 ow(i)= 20.032764635525957
control, rw(i)= 5.133347733327677 ow(i)= 25.666738666638388
```

Table including **rw(i)** and **ow(i)** for the most significant terms present in the top 10 documents based on ‘**drug legalization benefits**’ query.

term - t(i)	**r(i) out of 10**	**n(i)**	**rw(i)**	**ow(i)**
opinion	6	27988	6.20129725279	37.2077835168
current	5	58228	5.43599085675	27.1799542838
time	6	360476	2.42668099496	14.5600859698
addict	4	2320	8.74576839667	34.9830735867
alcohol	3	5067	7.84981983959	23.5315593372
law	6	59409	5.37963549005	32.2778129403
citi	2	72151	4.83636934007	9.67241475033
jail	1	7351	6.86176826771	6.86176826771
cost	4	81759	5.00819115888	20.0327646355
control	5	75691	5.13334773333	25.6667386666

Based on my calculated ow(i) values. I added some of these terms to my original query - **drug legalization benefits** to see how the ranked retrieval list has changed. For the purpose of this exercise I have selected the following terms to carry out query expansion.

1. jail
2. addict , law
3. cost, control, alcohol

Original query top 10	**Original query + “jail” top 10 documents**	**Original query + “addict”, “law” top 10 documents**	**Original query + “cost”, “control”, “alcohol” top 10 documents**
LA031289-0044	**LA031289-0044**	LA032590-0031	**LA031289-0044**
LA032590-0032	LA120289-0008	**LA032590-0032**	LA031990-0019
FBIS4-57566	LA091789-0139	LA103189-0060	FR940830-1-00101
FBIS4-67248	LA120889-0165	FBIS3-57998	LA120889-0165
FBIS3-9970	LA031990-0019	FBIS3-60059	LA120289-0008
FBIS3-21631	LA031889-0016	**LA031289-0044**	LA120289-0008
LA112089-0024	LA072089-0022	LA100889-0211	LA011090-0127
LA031990-0020	FR941017-1-00038	FBIS4-45284	LA091690-0032
FT931-16994	LA020690-0100	FBIS4-32836	LA040389-0018
FR940112-2-00078	LA040389-0018	FBIS4-62695	FR940208-2-00127

Using expanded queries with ‘**drug legalization benefits’** topic returned a lot of new documents. The only one that was consistent and seen within all of the expansion queries was article **LA031289-0044.** The same document also decreased in rank, when added some terms to it. Document **LA031289-0044** went from being number 1 in unexpanded query to number 6 when I added **‘law’** and **‘addict’** to the original query.

Investigations using qrel data

For this question I’ll examine the retrieved ranked list to identify if there any relevant documents based on the QREL. For the purpose of this question I’ll examine the original ‘**drug legalization benefits’** and - **‘drug legalization benefits’ ‘addict’ ‘law’** query retrieved lists against TREC qrel files.

Original unexpanded query (1 = relevant, 0 = not relevant)

Ranked results		**Relevant?**	
	“Drug”, “legalization”, “benefits”	**BM25**	**qrel**
1	LA031289-0044	1	1
2	LA032590-0032	1	1
3	FBIS4-57566	0	0
4	FBIS4-67248	0	0
5	FBIS3-9970	0	1
6	FBIS3-21631	0	1
7	LA112089-0024	1	1
8	LA031990-0020	0	1
9	FT931-16994	0	0

10	FR940112-2-00078	0	0

Original unexpanded query + “addict”, “law” (1 = relevant, 0 = not relevant)

Ranked results		Relevant?	
	“Drug”, “legalization”, “benefits”,”addict”,”law”	**BM25**	**qrel**
1	LA032590-0031	1	1
2	LA032590-0032	0	1
3	LA103189-0060	0	0
4	FBIS3-57998	1	1
5	FBIS3-60059	1	1
6	LA031289-0044	1	1
7	LA100889-0211	0	1
8	FBIS4-45284	0	0
9	FBIS4-32836	0	0
10	FBIS4-62695	0	0

From carrying out this exercise it was clear to see that expanding queries doesn’t always return more relevant results.