

CMPE 493, Introduction to Information Retrieval

Assignment 1 - A Simple Document Retrieval System for Boolean Queries

Utku SARIDEDE
2010400189

I have used well defined comments in source code. The one who wants to execute my source code should install some packages mentioned in readme file.
Answers of the questions:

- i.) a.) 2.745.153
b.) 1.850.520
c.) 92.267
d.) 64.799

- e.) (u'will', 14654)
(u'its', 14724)
(u'from', 14939)
(u'that', 15189)
(u'is', 16755)
(u'pct', 17046)
(u'on', 17797)
(u'it', 18082)
(u'"s", 18438)
(u'dlrs', 20268)
(u'The', 24271)
(u'for', 25194)
(u'mln', 25524)
(u'a', 48369)
(u'in', 49949)
(u'said', 52887)
(u'and', 53410)
(u'to', 68539)
(u'of', 72194)
(u'the', 119923)
- f.) (u'price', 7362)
(u'trade', 7434)
(u'net', 7703)
(u'inc', 7737)
(u'u.s.', 7979)
(u'market', 8369)
(u'would', 9233)
(u'ct', 9413)
(u'share', 10092)
(u'billion', 10696)
(u'compani', 11335)
(u'bank', 12149)
(u'year', 13365)
(u'vs', 14827)
(u'pct', 17962)
(u'"s", 19283)
(u'reuter', 20013)
(u'dlr', 24442)
(u'mln', 26675)
(u'said', 53089)

ii.) I have used hash-map approach to create inverted index. Hash-map keeps unique words and some unique dates(1985, 11.18.2002 etc.) as “key”. Values of these keys are lists (known as array in some languages) which include the indexes of keys. All other stuffs are written in the source code as comment outs.

iii.) Conjunctive and Disjunctive search query results are as the followings respectively;

```
[ utku: ~/Dropbox/Cmpe493/Assignment1/reuters21578 ]$
[ utku: ~/Dropbox/Cmpe493/Assignment1/reuters21578 ]$ python assignment1.py
assignment1.py:56: UnicodeWarning: Unicode equal comparison failed to convert both arguments to Unicode - interpreting them as being unequal
newTokenList = [ i for i in oneToken if i not in stopWordsList ]
What do you want to use? (a)AND or (o)OR:
~ 0
How many words does your query have?:
~ 3
Enter the 1th word: is
Enter the 2th word: excluDe
Enter the 3th word: approach
Result of Your Search, Indexes of Words:
[13, 54, 80, 160, 164, 169, 288, 299, 308, 317, 340, 356, 362, 381, 388, 419, 420, 422, 435, 449, 458, 490, 504, 530, 538, 557, 581, 603, 640, 644, 646, 679, 697, 721, 73
1, 736, 743, 759, 837, 859, 925, 927, 950, 998, 1010, 1011, 1084, 1114, 1121, 1128, 1132, 1137, 1160, 1176, 1198, 1200, 1236, 1245, 1251, 1258, 1270, 1285, 1314, 1353, 13
64, 1367, 1414, 1428, 1449, 1454, 1459, 1553, 1585, 1638, 1639, 1643, 1650, 1689, 1861, 1872, 1878, 1918, 1965, 1967, 1990, 1993, 2028, 2040, 2048, 2050, 2051, 2142, 2212
, 2278, 2309, 2311, 2316, 2333, 2350, 2352, 2353, 2380, 2386, 2507, 2508, 2512, 2548, 2597, 2632, 2652, 2659, 2667, 2691, 2708, 2718, 2748, 2750, 2805, 2856, 2872, 2878,
3024, 3249, 3337, 3449, 3506, 3528, 3532, 3534, 3535, 3543, 3545, 3557, 3563, 3602, 3615, 3634, 3637, 3669, 3672, 3738, 3768, 3785, 3806, 3820, 3840, 3867, 3900, 3909, 39
46, 3994, 3995, 4005, 4012, 4016, 4039, 4100, 4101, 4109, 4111, 4129, 4152, 4155, 4168, 4216, 4223, 4239, 4290, 4292, 4304, 4346, 4376, 4414, 4475, 4476, 4508, 4519, 4554
, 4581, 4611, 4671, 4677, 4710, 4728, 4755, 4765, 4788, 4823, 4824, 4860, 4893, 4902, 4914, 4916, 4930, 4943, 4946, 4974, 5060, 5076, 5095, 5112, 5128, 5172, 5176, 5290,
5312, 5332, 5339, 5359, 5376, 5407, 5417, 5422, 5438, 5452, 5458, 5488, 5544, 5561, 5566, 5573, 5575, 5576, 5586, 5587, 5592, 5684, 5692, 5735, 5744, 5767, 5786, 5817, 58
30, 5925, 5931, 5941, 5956, 5993, 6014, 6047, 6052, 6073, 6105, 6138, 6153, 6172, 6179, 6181, 6213, 6217, 6248, 6257, 6293, 6309, 6318, 6398, 6400, 6434, 6447, 6466, 6471
, 6594, 6606, 6619, 6656, 6681, 6716, 6742, 6787, 6793, 6796, 6813, 6837, 6882, 6893, 6896, 6914, 6922, 6989, 7048, 7092, 7115, 7126, 7135, 7155, 7163, 7192, 7212, 7229,
7254, 7272, 7326, 7338, 7421, 7455, 7465, 7495, 7499, 7525, 7589, 7593, 7629, 7645, 7733, 7765, 7767, 7771, 7883, 7917, 7927, 7950, 7953, 7954, 7980, 8084, 8017, 8027, 80
30, 8060, 8091, 8097, 8098, 8189, 8190, 8203, 8229, 8262, 8300, 8308, 8314, 8391, 8394, 8400, 8421, 8424, 8439, 8444, 8447, 8457, 8563, 8598, 8623, 8706, 8746, 8750, 8756
, 8860, 8892, 8896, 8906, 8961, 8979, 8999, 9032, 9072, 9097, 9158, 9175, 9231, 9278, 9316, 9339, 9465, 9502, 9524, 9525, 9533, 9549, 9618, 9634, 9647, 9693, 9706, 9714,
9715, 9755, 9788, 9795, 9796, 9813, 9861, 9873, 9884, 9894, 9952, 9963, 9975, 9987, 10003, 10046, 10091, 10097, 10101, 10138, 10150, 10155, 10165, 10185, 10195, 10211, 10
212, 10247, 10253, 10268, 10280, 10282, 10309, 10314, 10375, 10381, 10393, 10406, 10519, 10561, 10599, 10601, 10643, 10644, 10689, 10695, 10697, 10760, 10771, 10775, 1077
8, 10846, 10854, 10855, 10919, 10951, 10973, 10974, 10980, 10999, 11031, 11034, 11038, 11053, 11109, 11145, 11190, 11198, 11224, 11255, 11279, 11292, 11301, 11302, 11320,
11326, 11331, 11338, 11368, 11453, 11479, 11549, 11566, 11569, 11630, 11633, 11649, 11685, 11690, 11715, 11733, 11749, 11753, 11754, 11758, 11773, 11781, 11823, 11878, 1
1898, 11904, 12011, 12017, 12033, 12107, 12138, 12150, 12153, 12158, 12166, 12192, 12196, 12209, 12240, 12248, 12258, 12409, 12420, 12431, 12438, 12552, 12562, 125
69, 12573, 12624, 12642, 12655, 12715, 12860, 12874, 12941, 12943, 12948, 13056, 13074, 13159, 13214, 13261, 13263, 13282, 13576, 13715, 13772, 13958, 13966, 14434, 14451
, 14674, 14698, 14713, 14728, 14730, 14749, 14853, 14891, 14982, 14983, 15032, 15040, 15063, 15067, 15103, 15110, 15122, 15200, 15205, 15210, 15307, 15328, 15375, 15445,
15450, 15452, 15497, 15667, 15674, 15692, 15704, 15738, 15768, 15784, 15834, 15851, 15854, 15875, 15929, 15950, 16022, 16023, 16030, 16051, 16108, 16213, 16218, 16221, 16
272, 16284, 16286, 16312, 16332, 16347, 16367, 16389, 16558, 16587, 16630, 16644, 16773, 16775, 16782, 16783, 16804, 16812, 16815, 16819, 16843, 16871, 16888, 16908, 1694
8, 17031, 17034, 17091, 17148, 17207, 17220, 17233, 17238, 17389, 17391, 17427, 17469, 17477, 17654, 17668, 17701, 17708, 17714, 17731, 17779, 17791, 17795, 17862, 17894,
17897, 17965, 17996, 18008, 18049, 18144, 18194, 18204, 18250, 18258, 18326, 18333, 18393, 18428, 18491, 18614, 18679, 18695, 18719, 18867, 18875, 18973, 18983, 19006, 1
9046, 19078, 19375, 19378, 19403, 19418, 19514, 19555, 19598, 19604, 19611, 19644, 19742, 19757, 19787, 19815, 19825, 19848, 19854, 19945, 19964, 20035, 20057, 20174, 201
76, 20191, 20237, 20308, 20302, 20462, 20468, 20476, 20497, 20529, 20546, 20586, 20632, 20639, 20794, 20868, 20869, 20915, 20938, 20947, 20972, 20976, 21034, 21142, 21158
, 21164, 21188, 21215, 21242, 21260, 21275, 21381, 21412, 21447, 21455, 21490]
[ utku: ~/Dropbox/Cmpe493/Assignment1/reuters21578 ]$
```

```
[ utku: ~/Dropbox/Cmpe493/Assignment1/reuters21578 ]$
[ utku: ~/Dropbox/Cmpe493/Assignment1/reuters21578 ]$ python assignment1.py
assignment1.py:56: UnicodeWarning: Unicode equal comparison failed to convert both arguments to Unicode - interpreting them as being unequal
newTokenList = [ i for i in oneToken if i not in stopWordsList ]
What do you want to use? (a)AND or (o)OR:
~ 0
How many words does your query have?:
~ 4
Enter the 1th word: to
Enter the 2th word: reTuER
Enter the 3th word: still
Enter the 4th word: inc
Result of Your Search, Indexes of Words:
[635, 799, 1018, 1836, 1919, 2371, 2375, 2475, 2491, 2554, 2618, 2629, 2668, 2906, 3161, 3275, 3340, 3790, 3910, 4229, 4871, 4878, 4944, 5127, 5302, 5319, 5400, 5626, 562
7, 5750, 5869, 6197, 6357, 6626, 7036, 7263, 7361, 7366, 7499, 7607, 7787, 7839, 7897, 7925, 8098, 8205, 8334, 8439, 8441, 8525, 8569, 8853, 8927, 9046, 9246, 9258, 9290,
9640, 9644, 9658, 9679, 9817, 9826, 10103, 10118, 10187, 10423, 10444, 10546, 10582, 10768, 10786, 10931, 11064, 11257, 11292, 11575, 11642, 11645, 11714, 11763, 11888,
12118, 12136, 12195, 12318, 12747, 13073, 13124, 13208, 13210, 13592, 13904, 14691, 15063, 15160, 15241, 15264, 15363, 15530, 15639, 15932, 15968, 15988, 16028, 16053, 16
072, 16163, 16216, 16316, 16358, 16733, 16861, 17099, 17167, 17363, 17416, 17433, 17495, 17562, 17713, 17776, 17810, 18058, 18168, 18792, 18938, 19089, 19136, 19289, 1939
5, 19848, 20740, 20772, 21261, 21340, 21482]
[ utku: ~/Dropbox/Cmpe493/Assignment1/reuters21578 ]$
```