Οικονομικό Πανεπιστήμιο Αθηνών

Τμήμα Πληροφορικής

Συστήματα Ανάκτησης Πληροφοριών

Εαρινό Εξάμηνο 2019-2020

Εργασία 2

Κωνσταντίνα Λιάγκου

3150092

Αρχικοποίηση Project:

Για να μην είναι μεγάλο το το παραδοτέο δεν υπάρχει το documents.txt

Βάζουμε την συλλογή μας IR2020, που είναι σε ένα documents.txt μέσα στο φάκελο docs (που υπάρχει μέσα στο project)

Μορφή Παραδοτέου :

DataRetrieval είναι φάκελος που περιέχει το project, trec_eval_Results περιέχει

όλα τα αποτελέσμετα του trec_eval.

Επίσης με βάζει τις υποδείξεις στην περιοχή συζητήσεων υπάρχει ο φάκελος my_results με k πρώτα ανακτηθέντα κείμενα, για k=20, 30, 50. Αν τρέξει το πρόγραμμα θα δημιουργηθούν όλα τα txt όσα χρειάζονταν για την υλοποίηση της εργασίας, μέσα στο φάκελο DataRetrieval. Για να τρέξει το εργαλείο trec_eval από την κλάση Cmd.java, πρέπει τα αρχεία αυτά που θα πράξει η κλάση Searcher.java, να αντιγραφτούν μέσα στο φάκελο trec_eval (δεν χρειάζονται αντιγραφή, άμα δεν αλλαχτεί κάτι στο κώδικα, γιατί τα έχω αφήσει μέσα για να τρέχουν όλα ομαλά)

Προσοχή στο φάκελο docs μέσα στο project βάζουμε το txt για τα ερωτήματα και το μετονομάζουμε σε queries.txt για να το βρίσκει το πρόγραμμα (το queries. έχω αφήσει μέσα για να τρέχουν όλα ομαλά)

Βασικά σημεία υλοποίησης:

Ο κώδικας παραμένει σχεδόν ίδιος με τον κώδικα της $1^{ης}$ εργασίας.

Παρακάτω αναλύονται τα κομμάτια που χρειάστηκαν αλλαγή προκειμένου να επεκταθούμε το ερώτημα μας με συνώνυμους όρους.

Αρχικά, στην κλάση SearchDemo, αλλάξαμε τον EnglishAnalyzer με έναν CustomAnalyzer όπου δημιουργούμε μία συνάρτηση customAnalyzerForQueryExpansion().

Βήμα-Βήμα δοκιμές μέχρι να φτάσουμε στα καλυτέρα μας αποτελέσματα:

Η συνάρτηση customAnalyzerForQueryExpansion(), πρωταρχικά έκανε τα ίδια με τον EnglishAnalyzer,συν να προσθέτει τους συνώνυμους όρους από το θησαυρό WordNet. Για να το πετύχει αυτό διαβάζει τα συνώνυμα από το wn_s.pl και του λέμε ότι το αρχείο είναι τύπου Wordnet, δηλαδή κάνε το Parse σαν wordnet, όπως ξέρει η Lucene. Και έπειτα προστίθεται το filter της κλάσης SynonymGraphFilterFactory. Τα αποτελέσματα δεν ήτανε αρκετά καλά σε σχέση με την 1^{η} εργασία.

Τα 20 πρώτα ανακτηθέντα

num_rel_ret	Q01	9
map	Q01	0.4395
num_rel_ret	Q02	3
map	Q02	0.0663
num_rel_ret	Q03	8
map	Q03	0.4322
num_rel_ret	Q04	3
map	Q04	0.0590
num_rel_ret	Q05	5
map	Q05	0.1207
num_rel_ret	Q06	0
map	Q06	0.0000
num_rel_ret	Q07	3
map	Q07	0.0906
num_rel_ret	Q08	9
map	Q08	0.4366
num_rel_ret	Q09	1
map	Q09	0.0238
num_rel_ret	Q10	1
map	Q10	0.0125
num_rel_ret	all	42
map	all	0.1681

Τα 30 πρώτα ανακτηθέντα

num_rel_ret	Q01	11
map	Q01	0.4915
num_rel_ret	Q02	3
map	Q02	0.0663
num_rel_ret	Q03	10
map	Q03	0.4789
num_rel_ret	Q04	3
map	Q04	0.0590
num_rel_ret	Q05	9
map	Q05	0.1892
num_rel_ret	Q06	0
map	Q06	0.0000
num_rel_ret	Q07	4
map	Q07	0.0990
num_rel_ret	Q08	10
map	Q08	0.4641
num_rel_ret	Q09	2
map	Q09	0.0272
num_rel_ret	Q10	1
map	Q10	0.0125
num_rel_ret	all	53
map	all	0.1888

Τα 50 πρώτα ανακτηθέντα

num_rel_ret	all	70
map	all	0.2139
map_cut_5	all	0.1012
map_cut_10	all	0.1390
map_cut_15	all	0.1444
map_cut_20	all	0.1681
map_cut_30	all	0.1888
map_cut_100	all	0.2139
map_cut_200	all	0.2139
map_cut_500	all	0.2139
map_cut_1000	all	0.2139

Υ.Γ αναλυτικά για τα 50 πρώτα ανακτηθέντα θα δειχτεί μόνο στα τελικά αποτελέσματα(για βολικοτητα χώρου)

Στην συνέχεια δοκιμάστηκε να αλλαχτεί ο Standard Tokinizer που χρησιμοποιεί ο EnglishAnalyzer για tokenizer και χρησιμοποιήθηκε στην θέση του ο WhitespaceTokinizerFactory.class. Με αυτή την αλλάγή τα ποσοστά ανέβηκαν αρκετά

Τα 20 πρώτα ανακτηθέντα

Ενδικτικά: map: 42->48

num_rel_ret: 0.1681->0.1838

num_rel_ret	Q01	9
map	Q01	0.4395
num_rel_ret	Q02	3
map	Q02	0.0663
num_rel_ret	Q03	8
map	Q03	0.4322
num_rel_ret	Q04	3
map	Q04	0.0590
num_rel_ret	Q05	5
map	Q05	0.1207
num_rel_ret	Q06	0
map	Q06	0.0000
num_rel_ret	Q07	3
map	Q07	0.0906
num_rel_ret	Q08	9
map	Q08	0.4366
num_rel_ret	Q09	7
map	Q09	0.1800
num_rel_ret	Q10	1
map	Q10	0.0125
num_rel_ret	all	48
map	all	0.1838

Τα 30 πρώτα ανακτηθέντα

map: 53->60

num_rel_ret : 0.1888->0.2071

num_rel_ret	Q01	11
map	Q01	0.4915
num_rel_ret	Q02	3
map	Q02	0.0663
num_rel_ret	Q03	10
map	Q03	0.4789
num_rel_ret	Q04	3
map	Q04	0.0590
num_rel_ret	Q05	9
map	Q05	0.1892
num_rel_ret	Q06	0
map	Q06	0.0000
num_rel_ret	Q07	4
map	Q07	0.0990
num_rel_ret	Q08	10
map	Q08	0.4641
num_rel_ret	Q09	9
map	Q09	0.2106
num_rel_ret	Q10	1
map	Q10	0.0125
num_rel_ret	all	60
map	all	0.2071

Τα 50 πρώτα ανακτηθέντα

Ενδικτικά: map:70->79

num_rel_ret: 0.2139->0.2370

num_rel_ret	all	79
map	all	0.2370
map_cut_5	all	0.1084
map_cut_10	all	0.1479
map_cut_15	all	0.1563
map_cut_20	all	0.1838
map_cut_30	all	0.2071
map_cut_100	all	0.2370
map_cut_200	all	0.2370
map_cut_500	all	0.2370
map_cut_1000	all	0.2370

Αφού τα αποτελέσματα ήτανε εκπλήκτικά καλύτερα, ήρθε η ιδέα να φτιάξουμε και έναν customAnalyzer με αντικατάσταση του Standard με whitespace για το Ευρετήριο, οπότε αλλάκτηκε και η κλάση IndexerDemo,αλλάζοντας τον English. Πραγματικά είδαμε και άλλη βελτίωση στα δεδομένα.

Τα 20 πρώτα ανακτηθέντα

Ενδικτικά: map: 48 -> 50

num_rel_ret: 0.1838 -> 0.2015

Τα 30 πρώτα ανακτηθέντα

map: 60 -> 66

num_rel_ret : 0.2071 ->0.2307

num_rel_ret	Q01	9
map	Q01	0.4646
num_rel_ret	Q02	3
map	Q02	0.0873
num_rel_ret	Q03	9
map	Q03	0.4634
num_rel_ret	Q04	3
map	Q04	0.0643
num_rel_ret	Q05	6
map	Q05	0.1796
num_rel_ret	Q06	1
map	Q06	0.0088
num_rel_ret	Q07	2
map	Q07	0.0764
num_rel_ret	Q08	8
map	Q08	0.4411
num_rel_ret	Q09	7
map	Q09	0.1968
num_rel_ret	Q10	2
map	Q10	0.0325
num_rel_ret	all	50
map	all	0.2015

num_rel_ret	Q01	11
map	Q01	0.5140
num_rel_ret	Q02	3
map	Q02	0.0873
num_rel_ret	Q03	10
map	Q03	0.4958
num_rel_ret	Q04	4
map	Q04	0.0753
num_rel_ret	Q05	10
map	Q05	0.2589
num_rel_ret	Q06	1
map	Q06	0.0088
num_rel_ret	Q07	7
map	Q07	0.1367
num_rel_ret	Q08	9
map	Q08	0.4690
num_rel_ret	Q09	9
map	Q09	0.2289
num_rel_ret	Q10	2
map	Q10	0.0325
num_rel_ret	all	66
map	all	0.2307

num_rel_ret	all	78
map	all	0.2503
map_cut_5	all	0.1268
map_cut_10	all	0.1592
map_cut_15	all	0.1813
map_cut_20	all	0.2015
map_cut_30	all	0.2323
map_cut_100	all	0.2503
map_cut_200	all	0.2503
map_cut_500	all	0.2503
map_cut_1000	all	0.2503

Τα 50 πρώτα ανακτηθέντα

Ενδικτικά: map:79 ->78

num_rel_ret: 0.2370->0.2503

(Υ.Γ μπορεί να βλέπουμε ότι μόνο στα 50 χάνει ένα στο map, αλλά πάνω βρίσκει περισσότερα, και κυρίως ότι στα 50 ανακτηθέντα αυξάνονται τα σχετικά σε σχέση στα ανακτηθέντα μη σχετικά που είναι αρκετά σημαντικό ακόμα και αν χάνει ένα)

Στην συνέχεια δοκιμάστηκαν αρκετά Filter που δεν μας βοήθησαν, μας κρατάγανε το ποσοστό είτε σταθερό είτε μας το μείωναν. Ενδεικτικά, θα αναφερθούν τα πιο σημαντικά, που δεν είναι σαφές γιατί δεν δουλεψαν,ενώ θα έπρεπε. Αρχικά, δοκιμάστηκε ο LetterTokinizer, όπου
χωρίζει τους χαρακτήρες εισόδου σε όρους στα σημεία που δεν έχουν γράμμα (java.lang.Character.isLetter()).

Δηλαδή θεωρεί ότι είναι συνεχόμενες ακολουθίες από γράμματα είναι όροι.

Στην δικιά μας περίπτωση αυτό θα έπρεπε να δουλέψει δεδομένου ότι στα ερωτήματα έχουμε και παύλες π.χ. as-a…. Όταν χωρίζουμε με Whitespace, δεν θα τα χώριζε αυτά και θα τα έπαιρνε σαν ένα ενιαίο όρο ενώ είναι δυο ξεχωριστεί. Με το LetterTokinizer τα έπαιρνε όντως ξεχωριστά, αλλά με τα ερωτήματα που έχουμε δεν βοήθαγε καθόλου.

Με παρόμοια νοοτροπία δοκιμάστηκε και το hyphematedWordFilter, εξίσου δεν υπήρχε βελτίωση και παρέμεναν σταθερά.

Καθώς και το WordDelimiterFilter, που εκτός να χωρίζει τις λέξεις σε παύλες , διαχωρίζει και λέξεις με αριθμούς και διαχωρίζει και δυο λέξεις αν είναι ενωμένες και η δευτερη έχει κεφαλαίο.

Παράδειγματα από το documentation της:

Wi-Fi => Wi, Fi

PowerShot => Power, Shot

SD500 => SD, 500

Επιπρόσθετα, θα πρέπει να σημειωθεί ότι το WordNet δεν βοηθάει σε κάποιες περιπτώσεις. Όταν για παράδειγμα έχουμε ορολογία όπως το Big Data, δεν θέλουμε να βρούμε συνώνυμα του Big, γιατί το Big Data, δεν θα το λέγαμε ποτέ αλλιώς. Κάτι το οποίο γίνεται εύκολα αντιληπτό αν βγάλεις την λέξει Big, όπου τα ποσοστά αυξάνονται στα 50 ανακτηθέντα από 78 σε 83. Κάτι τέτοιο όμως θέλει κάποιο ειδικό να ορίσει ποιες λέξεις είναι όροι και αυτό δεν γίνεται με αυτοματοποιημένο τρόπο, συνεπώς δεν συμπεριλήφθηκε στην εργασία.

Επιπλέων, θα παρατηρήσουμε ότι παίζει αρκετά μεγάλο ρόλο ποτέ θα βρει τα συνώνυμα, επειδή το WordNet έχει τους όρους στον ενικό και όχι με stemming. Αναλυτικότερα, αν βρει τα συνώνυμα πριν κάνεις stemming θα βρει συνώνυμα για την λέξει community, ενώ αδυνατή μετά το stemming να βρει την λέξει comm. Από την άλλη πλευρά δεν μπορεί να βρει λέξεις τύπου networks, γιατί έχουν κατάληξει, ενώ βρίσκει μετά το stemming της λέξεις word.

(Υ.Γ Το δεύτερο παράδειγμα είναι και ένας από τους λόγους που στην αρχή που βγάλαμε τα συνώνυμα είχαμε περισσότερα μη σχετικά σε σχέση με το $1^{ο}$ παραδοτέο, μιας και έβαζε λέξεις συνώνυμες άκυρες από την αρχική αφού είχε υποστεί stemming και μπορεί να άλλαζε η σημασία της λέξεις.)

Παρότι προσπαθήσαμε να λύσουμε αυτό το θέμα με πάρα πολλούς τρόπους, να αλλάξουμε τα συνώνυμα τα βρίσκει πριν το stemming, ακόμα και να κάνουμε stemming πάνω στο αρχείο του WordNet. Όλες οι προσπάθειες πήγαν χαμένες και δεν βρέθηκε κάποια λύση γι΄αυτό το προβλημα.

Καλυτέρα αποτελέσματα:

Για να πετύχουμε τα μέγιστα μας αποτελέσματα χρειάστηκε να πειράξουμε ένα στοιχείο από τη κλάση SynonymGraphFilterFactory. Επειδή όμως η κλάση είναι read only, χρειάστηκε να αντιγραφτεί ολόκληρη σε μέσα στο φάκελο utils και να ονομαστεί MySynonym, ώστε να καλεστεί με αυτό το όνομα και να μην μπερδευτεί με την original. Αυτό που αλλάχτηκε είναι στην στη γραμμή 44 το expand που από False έγινε True. Με αυτό τον τρόπο του λέμε να δίνει μόνο το πρώτο πιο σχετικό συνώνυμο και όχι όλα. Έτσι του μειώνουμε το εύρος για να μην παίρνει σημασιολογικά άσχετους όρους. Και αυτό ξεπέρασε τα score και της $1^{η}$ εργασία. Και πραγματικά βλέπουμε τώρα το όφελος να χρησιμοποιήσουμε το WordNet για επέκταση του ερωτήματος σε σχέση να μην υπήρχαν οι συνώνυμη όροι όπως στην πρώτη εργασία. Παρακάτω υπάρχουν αναλυτικά τα αποτελέσματα από το τρέξιμο της Cmd.java για να δουμε να τα αποτελέσματα στο command prompt από το trec_eval.

Μας έχει ζητηθεί να δωθούν στο πραραδοτέο μόνο τα αποτελέσματα του trec_eval για map all,για τα κ=20,30,50 πρώτα ανακτηθέντα που υπάρχουν μέσα στο φάκελο trec_eval_Results. Για να δει κάποιος τα αποτελέσματα αναλυτικά sto cmd πρέπει να τρέξει την κλάση Cmd.java

Τα 20 πρώτα ανακτηθέντα

```
num_rel_ret             Q01     13
map                     Q01     0.7144
num_rel_ret             Q02     3
map                     Q02     0.2024
num_rel_ret             Q03     9
map                     Q03     0.4634
num_rel_ret             Q04     3
map                     Q04     0.0659
num_rel_ret             Q05     6
map                     Q05     0.1796
num_rel_ret             Q06     2
map                     Q06     0.0877
num_rel_ret             Q07     5
map                     Q07     0.1166
num_rel_ret             Q08     9
map                     Q08     0.6107
num_rel_ret             Q09     7
map                     Q09     0.1968
num_rel_ret             Q10     2
map                     Q10     0.1000
num_rel_ret             all     59
map                     all     0.2737
```

Τα 30 πρώτα ανακτηθέντα

```
num_rel_ret             Q01     16
map                     Q01     0.8251
num_rel_ret             Q02     3
map                     Q02     0.2024
num_rel_ret             Q03     10
map                     Q03     0.4958
num_rel_ret             Q04     4
map                     Q04     0.0789
num_rel_ret             Q05     10
map                     Q05     0.2589
num_rel_ret             Q06     2
map                     Q06     0.0877
num_rel_ret             Q07     8
map                     Q07     0.1668
num_rel_ret             Q08     11
map                     Q08     0.6673
num_rel_ret             Q09     9
map                     Q09     0.2289
num_rel_ret             Q10     3
map                     Q10     0.1115
num_rel_ret             all     76
map                     all     0.3123
```

Παρακάτω αναλυτικά τα αποτελέσματα για τα 50 πρώτα ανακτηθέντα:

Τα 50 πρώτα ανακτηθέντα:

```
num_rel_ret             Q01     16
map                     Q01     0.8251
map_cut_5               Q01     0.2941
map_cut_10              Q01     0.5235
map_cut_15              Q01     0.6694
map_cut_20              Q01     0.7144
map_cut_30              Q01     0.8251
map_cut_100             Q01     0.8251
map_cut_200             Q01     0.8251
map_cut_500             Q01     0.8251
map_cut_1000            Q01     0.8251
num_rel_ret             Q02     3
map                     Q02     0.2024
map_cut_5               Q02     0.1667
map_cut_10              Q02     0.2024
map_cut_15              Q02     0.2024
map_cut_20              Q02     0.2024
map_cut_30              Q02     0.2024
map_cut_100             Q02     0.2024
map_cut_200             Q02     0.2024
map_cut_500             Q02     0.2024
map_cut_1000            Q02     0.2024
num_rel_ret             Q03     13
map                     Q03     0.5581
map_cut_5               Q03     0.2536
map_cut_10              Q03     0.3607
map_cut_15              Q03     0.3940
map_cut_20              Q03     0.4634
map_cut_30              Q03     0.4958
map_cut_100             Q03     0.5581
map_cut_200             Q03     0.5581
map_cut_500             Q03     0.5581
map_cut_1000            Q03     0.5581
num_rel_ret             Q04     4
map                     Q04     0.0789
map_cut_5               Q04     0.0464
map_cut_10              Q04     0.0464
map_cut_15              Q04     0.0659
map_cut_20              Q04     0.0659
map_cut_30              Q04     0.0789
map_cut_100             Q04     0.0789
map_cut_200             Q04     0.0789
map_cut_500             Q04     0.0789
map_cut_1000            Q04     0.0789
num_rel_ret             Q05     12
map                     Q05     0.2992
map_cut_5               Q05     0.1198
map_cut_10              Q05     0.1198
map_cut_15              Q05     0.1599
map_cut_20              Q05     0.1796
map_cut_30              Q05     0.2589
map_cut_100             Q05     0.2992
map_cut_200             Q05     0.2992
map_cut_500             Q05     0.2992
map_cut_1000            Q05     0.2992
```

```
num_rel_ret             Q06     2
map                     Q06     0.0877
map_cut_5               Q06     0.0877
map_cut_10              Q06     0.0877
map_cut_15              Q06     0.0877
map_cut_20              Q06     0.0877
map_cut_30              Q06     0.0877
map_cut_100             Q06     0.0877
map_cut_200             Q06     0.0877
map_cut_500             Q06     0.0877
map_cut_1000            Q06     0.0877
num_rel_ret             Q07     10
map                     Q07     0.2028
map_cut_5               Q07     0.0625
map_cut_10              Q07     0.0625
map_cut_15              Q07     0.0863
map_cut_20              Q07     0.1166
map_cut_30              Q07     0.1668
map_cut_100             Q07     0.2028
map_cut_200             Q07     0.2028
map_cut_500             Q07     0.2028
map_cut_1000            Q07     0.2028
num_rel_ret             Q08     11
map                     Q08     0.6673
map_cut_5               Q08     0.3571
map_cut_10              Q08     0.5571
map_cut_15              Q08     0.6107
map_cut_20              Q08     0.6107
map_cut_30              Q08     0.6673
map_cut_100             Q08     0.6673
map_cut_200             Q08     0.6673
map_cut_500             Q08     0.6673
map_cut_1000            Q08     0.6673
num_rel_ret             Q09     10
map                     Q09     0.2448
map_cut_5               Q09     0.1310
map_cut_10              Q09     0.1310
map_cut_15              Q09     0.1615
map_cut_20              Q09     0.1968
map_cut_30              Q09     0.2448
map_cut_100             Q09     0.2448
map_cut_200             Q09     0.2448
map_cut_500             Q09     0.2448
map_cut_1000            Q09     0.2448
num_rel_ret             Q10     5
map                     Q10     0.1315
map_cut_5               Q10     0.1000
map_cut_10              Q10     0.1000
map_cut_15              Q10     0.1000
map_cut_20              Q10     0.1000
map_cut_30              Q10     0.1115
map_cut_100             Q10     0.1315
map_cut_200             Q10     0.1315
map_cut_500             Q10     0.1315
map_cut_1000            Q10     0.1315
num_rel_ret             all     86
map                     all     0.3298
map_cut_5               all     0.1619
map_cut_10              all     0.2191
map_cut_15              all     0.2538
map_cut_20              all     0.2737
map_cut_30              all     0.3139
```

Τέλος, για να κλίσουμε την εργασία παραθέτουμε σε διάγραμμα πόσο καλά τα πάει σε σύγκριση και με την $1^{η}$ εργασία και βλέπουμε, ότι όσο αυξάνονται τα k ανακτηθέντα αυξάνεται και το MAP