

**Οικονομικό Πανεπιστήμιο Αθηνών**  
**Τμήμα Πληροφορικής**

**Συστήματα Ανάκτησης Πληροφοριών**

**Εαρινό Εξάμηνο 2019-2020**

**Εργασία 1**

**Κωνσταντίνα Λιάγκου**

**3150092**

- Αρχικοποίηση Project:

(για να μην είναι μεγάλο το το παραδοτέο δεν υπάρχουν documents.txt, queries.txt)

Βάζουμε την συλλογή μας IR2020, που είναι σε ένα documents.txt μέσα στο φάκελο docs (που υπάρχει μέσα στο project), καθώς και στον ίδιο φάκελο βάζουμε και το queries.txt, με τα queries με βάση τα οποία θέλουμε να κάνουμε την αναζήτηση.

- Μορφή Παραδοτέου

DataRetrieval είναι φάκελος που περιέχει το project, trec\_eval\_Results περιέχει όλα τα αποτελέσματα του trec\_eval, το report αυτό, 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.

- Βασικά σημεία υλοποίησης:

Στην πρώτη Εργασία, φτιάξαμε ένα πρόγραμμα για την ανάκτηση κειμένων από την συλλογή IR2020 που έχει 18.316 κείμενα, η οποία βρίσκονταν σε ένα αρχείο documents.txt και δημιουργήσαμε μια μηχανή αναζήτησης που μας απαντά σε κάποια queries που μας δίνονται σε txt μορφή

Για να το πετύχουμε χρησιμοποιήσαμε την Lucene, που είναι εργαλείο ανακτήσεις. Συγκεκριμένα είναι μια βιβλιοθήκη που μας παρέχει έτοιμα εργαλεία για ευρετηρίαση, αναζήτηση και επεξεργαστώ της συλλογής

Πρωταρχικά, για να προεπεξεργαστούμε τη συλλογή (αρχείο documents.txt) προκειμένου να είναι σε κατάλληλη μορφή για να χρησιμοποιηθεί από τη μηχανή αναζήτησης Lucene κάναμε τα παρακάτω βήματα.

Αρχικά, περνάμε το μονοπάτι που υπάρχει το document.txt σε string προκειμένου να το δώσουμε στο TXTParsing (class) και να το διαβάσει και αποθήκευση σε μία μεταβλητή txt\_file τύπου string. Κάθε κείμενο εμφανίζει συγκεκριμένη δομή, που αποτυπώνεται στην class MyDoc, και φαίνεται ότι κάθε κείμενο έχει ένα docID, τίτλο και κυρίως κείμενο. Η TXTParsing με σκοπό να βρεί στο txt όλα τα διαφορετικά κείμενα, κάνει ένα διαχωρισμό(split) κάθε φορά που εμφανίζεται «`///`» καθώς και όσα whitespace ακολουθούν μετά (την εντολή `\s+` την βρήκα από <https://javarevisited.blogspot.com/2016/10/how-to-split-string-in-java-by-whitespace-or-tabs.html>). Και αφού τώρα έχει βρει τα διαφορετικά κείμενα, πάει σε κάθε κείμενο να πάρει το docID, τίτλο και κυρίως κείμενο για να φτιάξει ένα αντικείμενο τύπου MyDoc. Συγκεκριμένα, παίρνει τον αριθμό με την βοήθεια της split, που κάνει μόνο όταν βρει πρώτη

αλλαγή γραμμής(\n) και το υπόλοιπο κείμενο που μένει το διαχωρίζει από τον τίτλο, στο κομμάτι πριν βρει «:». Συνεπώς, η THTParsing θα επιστρέψει μία λίστα, που θα έχει μέσα όλα τα κείμενα (τύπου MyDoc) της συλλογής ξεχωριστά.

Επιπλέον, δημιουργούμε ένα ευρετήριο από τη συλλογή χρησιμοποιώντας τη μηχανή αναζήτησης Lucene, με βάση της οδηγίες που δόθηκαν στο εργαστήριο. Επιγραμματικά, ορίζω το Directory που θα αποθηκευτεί το ευρετήριο, δηλαδή στο φάκελο index. Ορίζω τον EnglishAnalyzer σαν Analyzer, μιας και αποδίδει καλύτερα από άλλους(π.χ StandardAnalyzer) και οι συλλογή μας είναι στα αγγλικά. Ο Analyzer κάνει όλη την διαδικασία της κοινωνικοποιήσεων, όπως εξαγωγή tokens, stemming κτλ . Έπειτα χρησιμοποιούμε BM25Similarity, σαν συνάρτηση ομοιότητας, που είναι και default, άμα δεν την γράψαμε. Αφού γίνουν όλα αυτά τα δίνουμε στον IndexWriter που είναι υπεύθυνος για τη μετατροπή του κειμένου σε internal Lucene format, δηλαδή format που καταλαβαίνει η lucene). Για κάθε κείμενο, καλούμε μία μέθοδο indexDoc ,που δημιουργεί τα 3 field και τα αποθηκεύει χωρίς να αναλυθεί στο ευρετήριο, για απλή ανάκτηση, καθώς και ένα επιπλέον, το contents Που έχει σαν text Searchable με βάζει ποια fields θα γίνει η αναζήτηση ,δηλαδή, τίτλο και κείμενο και όχι docID. Ειδικότερα, θα μπει στο ευρετήριο και θα γίνει και tokenization και analyze, αλλά δεν θα αποθηκευτεί. Τέλος, γίνονται add στο ευρετήριο.

Για να κάνω την αναζήτηση ακολουθώ ίδια διαδικασία για analyzer και διαβάσω τα queries.txt, με βάσει τα οποία θέλω να κάνω την αναζήτηση. Οι μόνες διαφορές με πριν είναι ότι έχουν δικιά τους class, που έχει τον αριθμό του κάθε query και το query. Και κάνουμε πάλι parsing, μόνο που αυτή την φορά χρησιμοποιούμε την MyQueryParsing, που ξεχωρίζει όλα τα query με βάσει πάλι «//\s+», αλλά πριν έχει σβήσει τα τελευταία, κάτω κάτω για να μην βρεί ένα query που να είναι το κενό. Έπειτα, κάθε query το διαχωρίζει από τον αριθμό του με βάσει την αλλαγή γραμμής (\r\n).

Στην SearchDemo, έχουμε επίσης έναν πίνακα που λέει πόσα συναφή πρώτα με το ερώτημα θα επιστραφούν και έχει k=5,10,15,20,30,50 και για καθένα από τα αποτελέσματα, φτιάχνουμε ένα txt διαφορετικό(my\_results\_file5, my\_results\_file30, my\_results\_file50 κτλ)

Κλείνοντας, για κάθε ένα my\_results\_file, που είναι οι απαντήσεις μας, συγκρίνοντάς τις με τις σωστές απαντήσεις (αρχείο qrels.txt) χρησιμοποιώντας το εργαλείο αξιολόγησης trec\_eval και τα μέτρα αξιολόγησης MAP (mean average precision) και avgPre@k (μέση ακρίβεια στα k πρώτα ανακτηθέντα κείμενα) για k=5, 10, 15, 20.

Αναλυτικότερα, τα αρχεία my\_results\_file είναι διαμορφωμένα κατάλληλα με βάση το πρότυπο που δέχεται ως όρισμα το trec\_eval. Η αξιολόγηση στο trec\_eval έγινε δίνοντας της ως είσοδο, κάθε φορά ένα από τα τρία αρχεία που παράξαμε με τα δικά μας αποτελέσματα

και μπροστά το αρχείο με τα πραγματικά σωστά, το qrels.txt . Φαίνεται αναλυτικά παρακάτω η εντολή εκτέλεσης του trec\_Eval και τα αποτελέσματα που επιστρέφονται.

Για τα μέτρα αξιολόγησης ,επέστρεψε:

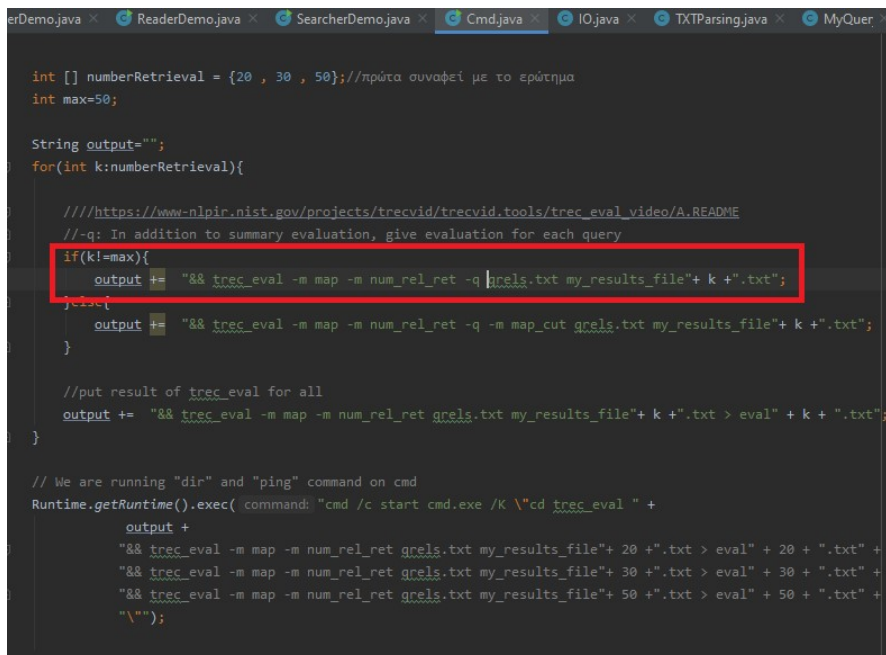
MAP (mean average precision) και avgPre@k (μέση ακρίβεια στα k πρώτα ανακτηθέντα κείμενα) για k=5, 10, 15, 20.

Στην περίπτωση μας το map@k (στα k πρώτα ανακτηθέντα) είναι ίδιο με το AvgPre@k για ένα σύστημα υπό αξιολόγηση, δηλαδή (MAP = AvgPre@k). Αν έχεις πολλά συστήματα(έστω r) είναι ο μέσος όρος των r επιμέρους AvgPre@k των συστημάτων.

Μέσα στο project υπάρχει μια class Cmd.java, η οποία με αυτόματο τρόπο τρέχει το trec\_eval και μας επιστρέφει όλα όσα ζητούνται από την εργασία.

Προσοχή, για να τρέξει σωστά αυτή η κλάση, πρέπει να έχει μέσα ο φάκελος trec\_eval, τα results που παράγονται από την κλάση SearcherDemo(φτιάχνονται μέσα στο φακελο Data Retrieval),καθώς επίσης να έχει μέσα και το qrels.txt. Προκειμένου να μην υπάρχει κάποια μπέρδεμα, έχω αφήσει όλα τα αρχεία μέσα στο φάκελο trec\_eval,οπότε η κλάση τρέχει κανονικά

Αναλυτικότερα, γράφουμε τις εντολές σε ένα string, το πρώτο απλά μας εμφανίζει στο cmd τις **για όλα τα query ξεχωριστά** για k=20,30 πρώτα ανακτηθέντα, με την **εντολή -q** όπου την βρήκα από το site που αναγράφεται και στον κώδικα



```
erDemo.java x ReaderDemo.java x SearcherDemo.java x Cmd.java x IO.java x TXTParsing.java x MyQuer x

int [] numberRetrieval = {20 , 30 , 50}; //πρώτα συναφεί με το ερώτημα
int max=50;

String output="";
for(int k:numberRetrieval){

    ///https://www-nlpir.nist.gov/projects/trecvid/trecvid.tools/trec_eval_video/A.README
    //-q: In addition to summary evaluation, give evaluation for each query
    if(k!=max){
        output += "&& trec_eval -m map -m num_rel_ret -q qrels.txt my_results_file"+ k +".txt";
    }else{
        output += "&& trec_eval -m map -m num_rel_ret -m map_cut qrels.txt my_results_file"+ k +".txt";
    }

    //put result of trec_eval for all
    output += "&& trec_eval -m map -m num_rel_ret qrels.txt my_results_file"+ k +".txt > eval" + k + ".txt";
}

// We are running "dir" and "ping" command on cmd
Runtime.getRuntime().exec( command: "cmd /c start cmd.exe /K \"cd trec_eval \" +
    output +
    "&& trec_eval -m map -m num_rel_ret qrels.txt my_results_file"+ 20 +".txt > eval" + 20 + ".txt" +
    "&& trec_eval -m map -m num_rel_ret qrels.txt my_results_file"+ 30 +".txt > eval" + 30 + ".txt" +
    "&& trec_eval -m map -m num_rel_ret qrels.txt my_results_file"+ 50 +".txt > eval" + 50 + ".txt" +
    "\\");
```

Για  $k=50$  χρησιμοποιούμε και την `cut_map`, ώστε να δούμε τα αποτελέσματα για `avgPre@k` (μέση ακρίβεια στα  $k$  πρώτα ανακτηθέντα κείμενα) για  $k=5, 10, 15, 20$ .

```
erDemo.java x ReaderDemo.java x SearcherDemo.java x Cmd.java x IO.java x TXTParsing.java x MyQuer x

int [] numberRetrieval = {20 , 30 , 50}; //πρώτα συναφεί με το ερώτημα
int max=50;

String output="";
for(int k:numberRetrieval){

    ///https://www-nlpir.nist.gov/projects/trecvid/trecvid.tools/trec_eval_video/A.README
    //-q: In addition to summary evaluation, give evaluation for each query
    if(k!=max){
        output += "&& trec_eval -m map -m num_rel_ret -q grels.txt my_results_file"+ k + ".txt";
    }else{
        output += "&& trec_eval -m map -m num_rel_ret -q -m map_cut grels.txt my_results_file"+ k + ".txt";
    }

    //put result of trec_eval for all
    output += "&& trec_eval -m map -m num_rel_ret grels.txt my_results_file"+ k + ".txt > eval" + k + ".txt";
}

// We are running "dir" and "ping" command on cmd
Runtime.getRuntime().exec( command: "cmd /c start cmd.exe /K \\\"cd trec_eval \" +
    output +
    "&& trec_eval -m map -m num_rel_ret grels.txt my_results_file"+ 20 + ".txt > eval" + 20 + ".txt" +
    "&& trec_eval -m map -m num_rel_ret grels.txt my_results_file"+ 30 + ".txt > eval" + 30 + ".txt" +
    "&& trec_eval -m map -m num_rel_ret grels.txt my_results_file"+ 50 + ".txt > eval" + 50 + ".txt" +
    "\\\"");
```

Δίνουμε ακριβώς μέσω των παραμέτρων ποιες μετρικές αξιολόγησης θέλουμε (με την βοήθεια του `-m`). Εμείς θέλουμε μόνο, για να δούμε αν βρήκαμε σωστά τα αποτελέσματα της 1<sup>η</sup> φάσης με τα προτεινόμενα από τους υπευθύνους του μαθήματος. 1) Το `map` (`-m map`) 2) Το Relative Returned Documents (`-m num_rel_ret`), που όπως λέει και το όνομα τους, είναι ποσά σχετικά επιστράφηκαν. Τέλος, για να έχουμε σε αρχείο τα αποτελέσματα, τα βάζουμε σε ένα αρχείο `evalm.txt` (εντολή στο τέλος: `> eval.txt`) και φαίνεται η τελική εντολή, παρακάτω για τα  $m=20,30$  και 50 ανακτηθέντα.

```
erDemo.java x ReaderDemo.java x SearcherDemo.java x Cmd.java x IO.java x TXTParsing.java x MyQuer x

int [] numberRetrieval = {20 , 30 , 50}; //πρώτα συναφεί με το ερώτημα
int max=50;

String output="";
for(int k:numberRetrieval){

    ///https://www-nlpir.nist.gov/projects/trecvid/trecvid.tools/trec_eval_video/A.README
    //-q: In addition to summary evaluation, give evaluation for each query
    if(k!=max){
        output += "&& trec_eval -m map -m num_rel_ret -q grels.txt my_results_file"+ k + ".txt";
    }else{
        output += "&& trec_eval -m map -m num_rel_ret -q -m map_cut grels.txt my_results_file"+ k + ".txt";
    }

    //put result of trec_eval for all
    output += "&& trec_eval -m map -m num_rel_ret grels.txt my_results_file"+ k + ".txt > eval" + k + ".txt";
}

// We are running "dir" and "ping" command on cmd
Runtime.getRuntime().exec( command: "cmd /c start cmd.exe /K \\\"cd trec_eval \" +
    output +
    "&& trec_eval -m map -m num_rel_ret grels.txt my_results_file"+ 20 + ".txt > eval" + 20 + ".txt" +
    "&& trec_eval -m map -m num_rel_ret grels.txt my_results_file"+ 30 + ".txt > eval" + 30 + ".txt" +
    "&& trec_eval -m map -m num_rel_ret grels.txt my_results_file"+ 50 + ".txt > eval" + 50 + ".txt" +
    "\\\"");
```

Τέλος, περνάμε όλο αυτό το string στην Runtime που βρήκαμε από το παρακάτω site:

<https://www.geeksforgeeks.org/java-program-open-command-prompt-insert-commands/>

Η ποία τρέχει το cmd και όλες τις εντολές που του δώσαμε μέσω του string output

```
erDemo.java x ReaderDemo.java x SearcherDemo.java x Cmd.java x IO.java x TXTParsing.java x MyQuer x

int [] numberRetrieval = {20 , 30 , 50}; //πρώτα συναφεί με το ερώτημα
int max=50;

String output="";
for(int k:numberRetrieval){

    ///https://www.nlpir.nist.gov/projects/trecvid/trecvid.tools/trec_eval_video/A.README
    //-q: In addition to summary evaluation, give evaluation for each query
    if(k!=max){
        output += "&& trec_eval -m map -m num_rel_ret -q grels.txt my_results_file"+ k + ".txt";
    }else{
        output += "&& trec_eval -m map -m num_rel_ret -q -m map_cut grels.txt my_results_file"+ k + ".txt";
    }

    //put result of trec_eval for all
    output += "&& trec_eval -m map -m num_rel_ret grels.txt my_results_file"+ k + ".txt > eval"+ k + ".txt";
}

// We are running "dir" and "ping" command on cmd
Runtime.getRuntime().exec( command: "cmd /c start cmd.exe /K \"cd trec_eval \" +
    output +
    "&& trec_eval -m map -m num_rel_ret grels.txt my_results_file"+ 20 + ".txt > eval" + 20 + ".txt" +
    "&& trec_eval -m map -m num_rel_ret grels.txt my_results_file"+ 30 + ".txt > eval" + 30 + ".txt" +
    "&& trec_eval -m map -m num_rel_ret grels.txt my_results_file"+ 50 + ".txt > eval" + 50 + ".txt" +
    "\\");
```

Τα αποτελέσματα που βγάζει άμα τρέξει, είναι τα παρακάτω

Για k=20 (για κάθε query ξεχωριστά και για όλα μαζί)

```
1 [main] trec_eval 2380 find_fast_cwd: WARNING: Couldn't compute FAST_CWD pointer. Please report this problem to
the public mailing list cygwin@cygwin.com
num_rel_ret      Q01      12
map              Q01      0.5706
num_rel_ret      Q02       3
map              Q02      0.1746
num_rel_ret      Q03       8
map              Q03      0.4322
num_rel_ret      Q04       3
map              Q04      0.0607
num_rel_ret      Q05       5
map              Q05      0.1207
num_rel_ret      Q06       1
map              Q06      0.0263
num_rel_ret      Q07       3
map              Q07      0.0934
num_rel_ret      Q08      11
map              Q08      0.6929
num_rel_ret      Q09       5
map              Q09      0.1683
num_rel_ret      Q10       2
map              Q10      0.0722
num_rel_ret      all      53
map              all      0.2412
```

Για k=30 (για κάθε query ξεχωριστά και για όλα μαζί)

```
1 [main] trec_eval 8548 find_fast_cwd: WARNING: Couldn't compute FAST_CWD pointer. Please report this problem to
the public mailing list cygwin@cygwin.com
644 [main] trec_eval 3948 find_fast_cwd: WARNING: Couldn't compute FAST_CWD pointer. Please report this problem to
the public mailing list cygwin@cygwin.com
num_rel_ret      Q01      14
map              Q01      0.6473
num_rel_ret      Q02      3
map              Q02      0.1746
num_rel_ret      Q03      10
map              Q03      0.4789
num_rel_ret      Q04      3
map              Q04      0.0607
num_rel_ret      Q05      9
map              Q05      0.1892
num_rel_ret      Q06      2
map              Q06      0.0301
num_rel_ret      Q07      9
map              Q07      0.1868
num_rel_ret      Q08      11
map              Q08      0.6929
num_rel_ret      Q09      7
map              Q09      0.1926
num_rel_ret      Q10      3
map              Q10      0.0822
num_rel_ret      all      71
map              all      0.2735
```

Για k=50 (για κάθε query ξεχωριστά και για όλα μαζί)

Μαζί και τα map\_cut, εμείς κοιτάμε μόνο map\_cut\_x ,x=5,10,15,20

```
the public mailing list cygwin@cygwin.com
348 [main] trec_eval 7980 find_fast_cwd: WARNING: Couldn't compute FAST_CWD pointer. Please report this problem to
the public mailing list cygwin@cygwin.com
num_rel_ret      Q01      15
map              Q01      0.6741
map_cut_5        Q01      0.1698
map_cut_10       Q01      0.3857
map_cut_15       Q01      0.5265
map_cut_20       Q01      0.5706
map_cut_30       Q01      0.6473
map_cut_100      Q01      0.6741
map_cut_200      Q01      0.6741
map_cut_500      Q01      0.6741
map_cut_1000     Q01      0.6741
num_rel_ret      Q02      3
map              Q02      0.1746
map_cut_5        Q02      0.1389
map_cut_10       Q02      0.1746
map_cut_15       Q02      0.1746
map_cut_20       Q02      0.1746
map_cut_30       Q02      0.1746
map_cut_100      Q02      0.1746
map_cut_200      Q02      0.1746
map_cut_500      Q02      0.1746
map_cut_1000     Q02      0.1746
num_rel_ret      Q03      14
map              Q03      0.5687
map_cut_5        Q03      0.2536
map_cut_10       Q03      0.3743
map_cut_15       Q03      0.3743
map_cut_20       Q03      0.4322
map_cut_30       Q03      0.4789
map_cut_100      Q03      0.5687
map_cut_200      Q03      0.5687
map_cut_500      Q03      0.5687
map_cut_1000     Q03      0.5687
num_rel_ret      Q04      4
map              Q04      0.0694
map_cut_5        Q04      0.0464
map_cut_10       Q04      0.0464
map_cut_15       Q04      0.0607
map_cut_20       Q04      0.0607
map_cut_30       Q04      0.0607
map_cut_100      Q04      0.0694
map_cut_200      Q04      0.0694
map_cut_500      Q04      0.0694
map_cut_1000     Q04      0.0694
num_rel_ret      Q05      12
map              Q05      0.2401
map_cut_5        Q05      0.0896
map_cut_10       Q05      0.0896
map_cut_15       Q05      0.0896
map_cut_20       Q05      0.1207
map_cut_30       Q05      0.1892
map_cut_100      Q05      0.2401
map_cut_200      Q05      0.2401
map_cut_500      Q05      0.2401
map_cut_1000     Q05      0.2401
```



num_rel_ret	Q06	5
map	Q06	0.0445
map_cut_5	Q06	0.0263
map_cut_10	Q06	0.0263
map_cut_15	Q06	0.0263
map_cut_20	Q06	0.0263
map_cut_30	Q06	0.0301
map_cut_100	Q06	0.0445
map_cut_200	Q06	0.0445
map_cut_500	Q06	0.0445
map_cut_1000	Q06	0.0445
num_rel_ret	Q07	12
map	Q07	0.2382
map_cut_5	Q07	0.0625
map_cut_10	Q07	0.0764
map_cut_15	Q07	0.0934
map_cut_20	Q07	0.0934
map_cut_30	Q07	0.1868
map_cut_100	Q07	0.2382
map_cut_200	Q07	0.2382
map_cut_500	Q07	0.2382
map_cut_1000	Q07	0.2382
num_rel_ret	Q08	11
map	Q08	0.6929
map_cut_5	Q08	0.3571
map_cut_10	Q08	0.5714
map_cut_15	Q08	0.5714
map_cut_20	Q08	0.6929
map_cut_30	Q08	0.6929
map_cut_100	Q08	0.6929
map_cut_200	Q08	0.6929
map_cut_500	Q08	0.6929
map_cut_1000	Q08	0.6929
num_rel_ret	Q09	9
map	Q09	0.2111
map_cut_5	Q09	0.1310
map_cut_10	Q09	0.1500
map_cut_15	Q09	0.1683
map_cut_20	Q09	0.1683
map_cut_30	Q09	0.1926
map_cut_100	Q09	0.2111
map_cut_200	Q09	0.2111
map_cut_500	Q09	0.2111
map_cut_1000	Q09	0.2111
num_rel_ret	Q10	3
map	Q10	0.0822
map_cut_5	Q10	0.0500
map_cut_10	Q10	0.0722
map_cut_15	Q10	0.0722
map_cut_20	Q10	0.0722
map_cut_30	Q10	0.0822
map_cut_100	Q10	0.0822
map_cut_200	Q10	0.0822
map_cut_500	Q10	0.0822
map_cut_1000	Q10	0.0822

```
num_rel_ret      all      88
map              all      0.2996
map_cut_5        all      0.1325
map_cut_10       all      0.1967
map_cut_15       all      0.2157
map_cut_20       all      0.2412
map_cut_30       all      0.2735
map_cut_100      all      0.2996
map_cut_200      all      0.2996
map_cut_500      all      0.2996
map_cut_1000     all      0.2996
```

!!!Προσοχή, μας έχει ζητηθεί να δωθούν στο παραδοτέο 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 !!!

Τέλος, για να κλήσουμε την εργασία παραθέτουμε σε διάγραμμα πόσο καλά τα πάει και βλέπουμε, ότι όσο αυξάνονται τα κ ανακτηθέντα αυξάνεται και το Map

