Αναφορά 1ης φάσης Εργασίας

Αλέξανδρος Βεντούρας AM: 3160013 email: alvent98@gmail.com

Στην παρούσα αναφορά περιγράφονται τα βήματα που ακολουθήθηκαν κατά την πρώτη φάση (Baseline) της εργασίας του μαθήματος «Συστήματα Ανάκτησης Πληροφοριών».

Α. Πηγές κώδικα:

Οι πηγές που χρησιμοποιήθηκαν σε αυτό το στάδιο, είναι πρωτίστως το overview του Documentation της συγκεκριμένης έκδοσης του Lucene[1] (χρησιμοποιήθηκε η έκδοση 7.7.2.), το οποίο περιέχει δύο αρχεία, τα IndexFiles.java και SearchFiles.java, που αποτέλεσαν τη βάση για τη συγγραφή των παραδιδόμενων προγραμμάτων. Επιπλέον, αξιοποιήθηκε και ένα βίντεο στο youtube[2], βάσει των οδηγιών του οποίου προστέθηκαν ως external JAR's ορισμένα αρχεία της βιβλιοθήκης Lucene.

Β. Περιγραφή υλοποίησης τριών πρώτων βημάτων – ανάλυση κώδικα:

Όσον αφορά τα τρία πρώτα βήματα, έγινε parsing του εγγράφου documents.txt κατά τον τρόπο που περιγράφεται στις ακόλουθες παραγράφους.

Αρχικά (αναφορικά με την δημιουργία του ευρετηρίου) αποθηκεύεται η πρώτη σειρά (σαν StringField με property "id"), η οποία είναι προφανές ότι περιέχει το id του πρώτου document (γραμμές 42 – 44). Έπειτα, γινόταν ανάγνωση κάθε επόμενης σειράς (γραμμή 46), και αν αυτή η σειρά περιείχε τρεις καθέτους συνεχόμενα (δηλαδή την ακολουθία '///' χωρίς τα εισαγωγικά), τότε αυτό σήμαινε ότι από την επόμενη γραμμή ξεκινάει καινούργιο document. Συνεπώς, αποθηκευόταν οι ήδη διαβασμένες γραμμές σε ένα TextField με property "contents" (γραμμή 52), και αν υπήρχε ήδη το document, τότε έπρεπε να γίνει append σε αυτό το νέο property, αλλιώς αν δεν υπήρχε, προστίθεντο στον writer ως νέο document (γραμμή 59). Αν προς το παρόν μόνο create γίνεται, η περίπτωση του append παρέμεινε, μήπως χρειαστεί σε επόμενη φάση της εργασίας. Έπειτα, διαβάζεται η επόμενη γραμμή και αποθηκεύεται το id του επόμενου κειμένου (γραμμές 80-82).

Σε διαφορετική περίπτωση, δηλαδή αν η διαβασμένη γραμμή δεν περιείχε την ακολουθία των τριπλών καθέτων, απλά περνούσαμε στην ανάγνωση της επόμενης γραμμής ,αφού πρώτα είχε προστεθεί η τρέχουσα γραμμή σε ένα string (γραμμή 84).

Τα παραπάνω γίνονται στην μέθοδο indexDocument της κλάσης IndexCreator. Στην main του συγκεκριμένου αρχείου (IndexCreator.java), καταχωρείται το path του txt που περιέχει τα κείμενα, τα οποία περιέχουν τις ζητούμενες πληροφορίες (γραμμή 99). Έπειτα, «ανοίγεται» το directory το οποίο περιέχει το txt με τις πληροφορίες (γραμμή 106), και αρχικοποιείται ο αναλυτής (Analyzer), ως στιγμιότυπο της ειδικότερης κλάσης EnglishAnalyzer(γραμμή 107). Επιλέχθηκε ο συγκεκριμένος Analyzer[3], προκειμένου να

[1] https://lucene.apache.org/core/7_7_2/demo/overview-summary.html (τελευταία επίσκεψη στις 28-4-2020)
[2] https://www.youtube.com/watch?v=pVDVURw_AJQ (τελευταία επίσκεψη στις 28-4-2020)
[3] Βάσει του συγκεκριμένου άρθρου, το οποίο αναλύει τα features κάθε διαθέσιμου Analyzer στη Lucene: https://www.baeldung.com/lucene-analyzers (τελευταία επίσκεψη στις 29-4-2020)

εκμεταλλευτούμε το γεγονός ότι εκτελεί αυτόματα την απαραίτητη γλωσσολογική επεξεργασία, όπως την αφαίρεση των stopwords, ή την αποκοπή των παραγωγικών καταλήξεων, με τη βοήθεια του Porter Stemmer.

Έπειτα, δημιουργείται ένας εγγραφέας ευρετηρίου (IndexWriter – γραμμή 121), που κατ' ουσίαν παραμετροποιείται με todirectory, το οποίο περιέχει το path του txt, καθώς και τον IndexWriterConfig, που εμπερικλείει τον EnglishAnalyzer, τον οποίο δημιουργήσαμε προηγουμένως. Τέλος, εκτελείται η μέθοδος indexDocument (γραμμή 124), που περιγράφεται παραπάνω.

Όσον αφορά την αναζήτηση, αυτή εκτελείται μέσω της μεθόδου search της κλάσης Searcher, του ομώνυμου αρχείου. Σε αυτή, αρχικά μέσω της κλήσης της μεθόδου search της κλάσης IndexSearcher (του Lucene), γίνεται η αναζήτηση ενός από τα queries, στο ευρετήριο που έχει ήδη δημιουργηθεί (γραμμή 31).

Έπειτα, αφού αποθηκευτούν τα αποτελέσματα της αναζήτησης (γραμμή 32), αποθηκεύονται σε ένα αρχείο τα στοιχεία κάθε document που επιτυγχάνει πάνω από ένα συγκεκριμένο αριθμό ευρέσεων (γραμμές 50-52 και 54).

Όσον αφορά το parsing των queries, αυτό γίνεται στη μέθοδο getQueries του αρχείου Searcher. Σε αυτή, διαβάζεται γραμμή-γραμμή το αρχείο που τα περιέχει, και αποθηκεύονται μόνο οι γραμμές που περιέχουν μόνο γράμματα ή και παύλες, κάθε μία γραμμή σε μία ξεχωριστή λίστα από Strings (γραμμές 69-70).

Έπειτα, στη main του συγκεκριμένου αρχείου, δημιουργείται το αρχείο στο οποίο θα αποθηκευτούν τα αποτελέσματα, αφού ανακτηθούν από το αρχείο στο οποίο είναι γραμμένα (γραμμή 100), και πάλι με τη χρήση του EnglishAnalyzer αυτά επεξεργάζονται κατάλληλα, προκειμένου οι λέξεις που περιέχουν να είναι στην ίδια μορφή με αυτές των documents (γραμμή 113). Έπειτα εκτελείται η μέθοδος search, της οποίας η λειτουργία έχει ήδη αναλυθεί. Η δημιουργία των τριών αρχείων, results20.txt, results30.txt και results50.txt,έγινε με την αλλαγή της τιμής της μεταβλητής hitsInPage (γραμμή 102), καθώς και με την αλλαγή του ονόματος του εκάστοτε αρχείου στις γραμμές 40 και 91.

Τέλος, είναι πολύ σημαντικό να τονιστεί ότι σε περίπτωση που μεταβληθεί το path των σχετικών αρχείων (εκτέλεση σε άλλον υπολογιστή κλπ), πρέπει να ενημερωθούν τα paths που υπάρχουν στις γραμμή 96 του αρχείου IndexCreator.java, και στις γραμμές 40, 91 και 100 του αρχείου Searcher.java, με τα αντίστοιχα paths στα οποία βρίσκονται τα αρχεία της συλλογής IR2020.

Γ. Περιγραφή τέταρτου βήματος (χρήση trec_eval):

Έπειτα από την εκτέλεση των τριών πρώτων βημάτων, μέσα από την χρήση της γραμμής εντολών, εκτελέστηκε η trec_eval, και παρήχθησαν τα αρχεία που συμπεριλαμβάνονται στο zip αρχείο, μέσα στον φάκελο txts, σύμφωνα με την υπόδειξη που υπάρχει στην διαφάνεια 8 του αρχείου trec_eval.pdf. Παρακάτω υπάρχουν τα screenshots από τις διάφορες εκτελέσεις της trec_eval, για κάθε μετρική που ζητήθηκε:

i) Average Precision (AvgPre@k):

```
C:\Users\a\Desktop\txts>trec_eval -q -m map_cut.5 qrels.txt results20.txt > eval_map_cut_05.txt
C:\Users\a\Desktop\txts>trec_eval -q -m map_cut.10 qrels.txt results20.txt > eval_map_cut_10.txt
C:\Users\a\Desktop\txts>trec_eval -q -m map_cut.15 qrels.txt results20.txt > eval_map_cut_15.txt
C:\Users\a\Desktop\txts>trec_eval -q -m map_cut.20 qrels.txt results20.txt > eval_map_cut_20.txt
```

ii) Relative Returned Documents

```
C:\Users\a\Desktop\txts>trec_eval -q -M 5 -m num_rel_ret qrels.txt results20.txt > eval_num_rel_ret_05.txt
C:\Users\a\Desktop\txts>trec_eval -q -M 10 -m num_rel_ret qrels.txt results20.txt > eval_num_rel_ret_10.txt
C:\Users\a\Desktop\txts>trec_eval -q -M 15 -m num_rel_ret qrels.txt results20.txt > eval_num_rel_ret_15.txt
C:\Users\a\Desktop\txts>trec_eval -q -m num_rel_ret qrels.txt results20.txt > eval_num_rel_ret_20.txt
C:\Users\a\Desktop\txts>trec_eval -q -m num_rel_ret qrels.txt results30.txt > eval_num_rel_ret_30.txt
C:\Users\a\Desktop\txts>trec_eval -q -m num_rel_ret qrels.txt results50.txt > eval_num_rel_ret_50.txt
```

iii) Mean Average Precision (MAP@k):

```
C:\Users\a\Desktop\txts>trec_eval -q -m map qrels.txt results20.txt > eval_map_20.txt
C:\Users\a\Desktop\txts>trec_eval -q -m map qrels.txt results30.txt > eval_map_30.txt
C:\Users\a\Desktop\txts>trec_eval -q -m map qrels.txt results50.txt > eval_map_50.txt
```

Στις επόμενες σελίδα περιλαμβάνεται ο πίνακας με τα περιεχόμενα των παραχθέντων αρχείων, καταπώς υποδείχτηκε.

Query ID:	**Total Returned Documents k:**	**Average Precision *AvgPre@k:***	**Relative Returned Documents:**	**Mean Average Precision *MAP@k:***
Q01	k = 5	0.1698	4	
	k = 10	0.3857	8	
	k = 15	0.5307	11	
	k = 20	0.5749	12	0.5749
	k = 30		14	0.6516
	k = 50		15	0.6792
Q02	k = 5	0.1389	2	
	k = 10	0.1746	3	
	k = 15	0.1746	3	
	k = 20	0.1746	3	0.1746
	k = 30		3	0.1746
	k = 50		3	0.1746
Q03	k = 5	0.2536	4	
	k = 10	0.3743	6	
	k = 15	0.3743	6	
	k = 20	0.4323	8	0.4323
	k = 30		11	0.5078
	k = 50		14	0.5745

Q04	k = 5	0.0464	2	
	k = 10	0.0464	2	
	k = 15	0.0464	2	
	k = 20	0.0598	3	0.0598
	k = 30		3	0.0598
	k = 50		4	0.0688
Q05	k = 5	0.0896	3	
	k = 10	0.0896	3	
	k = 15	0.0896	3	
	k = 20	0.1207	5	0.1207
	k = 30		9	0.1911
	k = 50		13	0.2619
Q06	k = 5	0.0263	1	
	k = 10	0.0263	1	
	k = 15	0.0263	1	
	k = 20	0.0263	1	0.0263
	k = 30		2	0.0304
	k = 50		4	0.0391
Q07	k = 5	0.0625	1	
	k = 10	0.0764	2	
	k = 15	0.0934	3	
	k = 20	0.0934	3	0.0934
	k = 30		9	0.1880
	k = 50		12	0.2383
Q08	k = 5	0.3571	5	
	k = 10	0.5714	8	
	k = 15	0.6143	9	
	k = 20	0.6982	11	0.6982
	k = 30		11	0.6982
	k = 50		11	0.6982
Q09	k = 5	0.1310	3	
	k = 10	0.1500	4	
	k = 15	0.1683	5	
	k = 20	0.1683	5	0.1683
	k = 30		7	0.1936
	k = 50		9	0.2126
Q10	k = 5	0.0500	1	
	k = 10	0.0750	2	
	k = 15	0.0750	2	
	k = 20	0.0750	2	0.0750
	k = 30		3	0.0853
	k = 50		3	0.0853
All	k = 5	0.1325	26	
	k = 10	0.1970	39	
	k = 15	0.2193	45	
	k = 20	0.2424	53	0.2424
	k = 30		72	0.2780
	k = 50		88	0.3033