Συστήματα ανάκτησης πληροφοριών

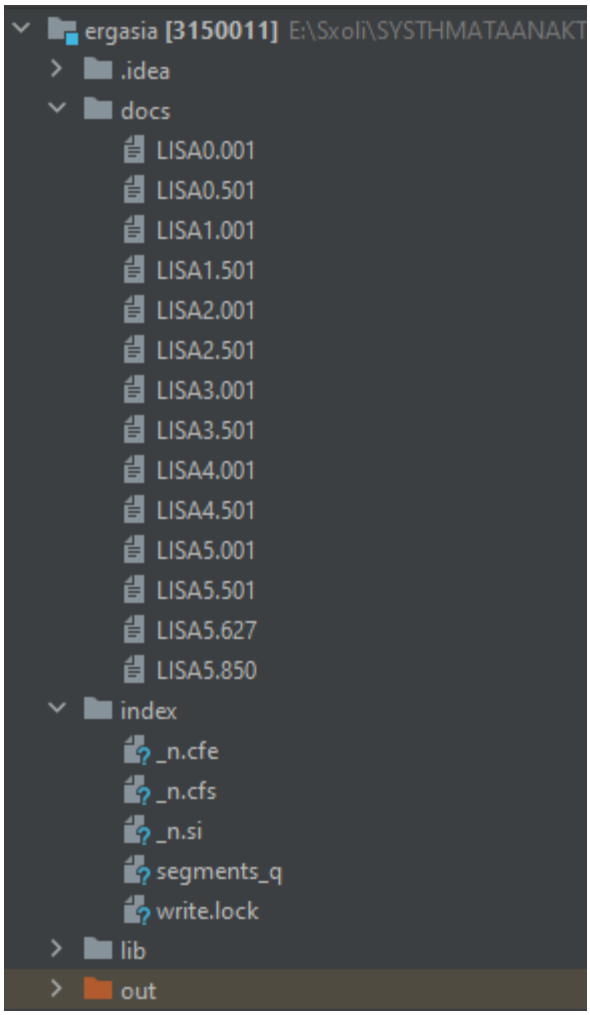
1η Φάση Προγραμματιστικής Εργασίας

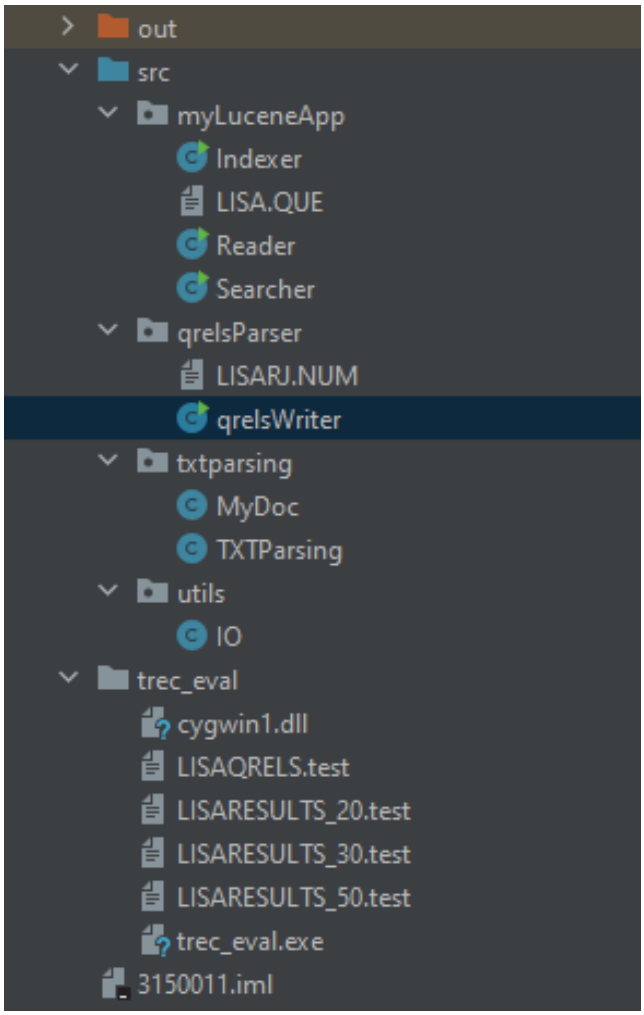
Βιβλιοθήκη LISA

Ον/μο: ΒΙΤΑΛΗΣ ΙΩΑΝΝΗΣ

ΑΜ: 3150011

Τελική Δομή:





Επεξεργασία LISARJ.NUM / LISA.REL

Αρχικά ασχολήθηκα με την μετατροπή του LISA.REL στην κατάλληλη μορφή για να διαβαστεί από το trec_eval, παρατηρώντας το LISA.REL είδα οτι μερικές εγγραφές ήταν λανθασμένα καταχώρημενες και οτι το αρχείο LISARJ.NUM παρείχε πιο ολοκληρωμένη πληροφορία.

Η μετατροπή επιτεύχθηκε με την δημιουργία του qrelsWriter, το οποίο διαβάζει το LISARJ.NUM, το μετατρέπει σε μορφή κατάλληλη για να διαβαστεί από το trec_eval και το αποθηκεύει στον φάκελο trec_eval για μελλοντική χρήση.

Δημιουργία Ευρετηρίου

Το διάβασμα το αρχείων της συλλογής γίνεται στον Indexer, στην μέθοδο readFiles επειδή η συλλογή LISA ήταν «σκορπισμένη» σε συνολικά 14 αρχεία.

Χρησιμοποιούμε τον EnglishAnalyzer() και ClassicSimilarity() και κάνουμε parsing καλώντας την μέθοδο parse(String file) του Class TXTParsing.

Η parse αντικαθιστά τα πολλαπλά «****» που χωρίζουν μεταξύ τους τα documents, με ένα μονο «*» και χρησιμοποιούμε αυτό για να κάνουμε split το αρχείο στα ξεχωριστά documents στον πίνακα docs[].

Ο πίνακας docs[] παρατηρούμε οτι στην πρώτη θέση(adoc[0]) έχει το ID και στην επόμενη γραμμή το Title του document, στην δεύτερη θέση(adoc[1]) έχει το κυρίως κείμενο του document.

Χωριζούμε το adoc[0] κάθε φορά που αλλάζει γραμμή, η πρώτη γραμμή περιέχει το ID του document και οι υπόλοιπες γραμμές περιέχουν το Title του document.

Το id είναι της μορφής «Document   3245», με replaceAll("[^\\d.]",""); αφαιρούμε μπροστά την λέξη Document που δεν έχει καμία χρησιμότητα και κραταμε μόνο το ID του κειμένου.

Εκτέλεση ερωτημάτων στο ευρετήριο

Η εκτέλεση των ερωτημάτων στο ευρέτηριο γίνεται στην Searcher, διαβάζει το index που δημιουργήθηκε από τον Indexer, δημιουργεί έναν IndexSearcher με similarity το ClassicSimilarity().

Για τα πλαίσια της εργασίας η μέθοδος search καλείτε 3 φορές προκειμένου να δημιουργηθούν 3 διαφορετικά αρχεία, ένα για k=20 ανακτηθέντα αρχεία, ένα για k=30 και ένα για k=50.

Δημιουργούμε έναν QueryParser με analyzer τον EnglishAnalyzer() που χρησιμοποιήσαμε και στον Indexer για την δημιουργία του index.

Διαβάζουμε όλα τα queries προς αξιολόγηση με την μέθοδο queryReader(), η οποία διαβάζει τον πρώτο αριθμό (query ID) και μετά διαβάζει όλες τις επόμενες γραμμές και τις προσθέτει σε

ένα String ref, αν η επόμενη γραμμή είναι αριθμός αποθηκεύει το String ref στο Arraylist που θα επιστραφεί και διαβάζει το επόμενο query. Στο τέλος επιστρέφει το Arraylist που περιέχει όλα τα Queries προς αξιολόγηση.

Η IndexSearcher καλεί την search για κάθε Query στο ArrayList και γράφει τα αποτελέσματα στο αρχείο LISARESULTS_K.text σε κατάλληλη μορφή για να χρησιμοποιηθεί αργότερα από το εργαλείο trec_eval.

Στο τέλος ο φάκελος trec_eval θα πρέπει να περιέχει τα εξής αρχεία

- LISAQRELS.test που δημιουργήθηκε από το πρόγραμμα qrelsWriter.
- LISARESULTS_20.test που δημιουργήθηκε από το πρόγραμμα Searcher για k=20 ανακτηθέντα κείμενα.
- LISARESULTS_30.test που δημιουργήθηκε από το πρόγραμμα Searcher για k=30 ανακτηθέντα κείμενα.
- LISARESULTS_50.test που δημιουργήθηκε από το πρόγραμμα Searcher για k=50 ανακτηθέντα κείμενα.

Αποτελέσματα trec_eval

Μέτρο Αξιολόγησης / **Κ πρώτα Ανακτηθέντα κείμενα**	**MAP (Mean Average Precision)**	**avgPre@k (Average Precision @k)**
K = 20	**MAP = 0.2039**	**P_05 = 0.2800** **P_10 = 0.1850** **P_15 = 0.1600** **P_20 = 0.1425**
K = 30	**MAP = 0.2191**	**P_05 = 0.2800** **P_10 = 0.1850** **P_15 = 0.1600** **P_20 = 0.1425**

K = 50	**MAP = 0.2256**	**P_05 = 0.2800** **P_10 = 0.1850** **P_15 = 0.1600** **P_20 = 0.1425**