Ανάκτηση Πληροφορίας

Εαρινό Εξάμηνο 2024

Εργασία: Μηχανή αναζήτησης επιστημονικών άρθρων

1: Περιγραφή δεδομένων και αρχικός σχεδιασμός

Ομάδα
Γκαβαρδίνα Αγγελική 4042
Αλέξανδρος Κόκκινος 4084

GitHub Link

https://github.com/aggelikigkavardina/info-retrieval-24-4042-4084

Επιλογή συλλογής εγγράφων και περιγραφή πεδίων αυτών.

Επιλέξαμε την συλλογή εγγράφων All NeurIPS (NIPS) Papers που μας προτάθηκε να χρησιμοποιήσουμε για την υλοποίηση της μηχανής αναζήτησης. Αρχικά κατεβάσαμε την συλλογή από την ιστοσελίδα Kaggle.com.

Η συλλογή έχει 2 αρχεία:

1. Το αρχείο authors που περιέχει τις στήλες source_id που περιέχει τα id του άρθρου το οποίο έχει δημοσιεύσει ο κάθε συγγραφέας, first_name που περιέχει το μικρό τους όνομα, last_name που περιέχει το επίθετό τους και institution που λέει για ποιόν φορέα εργάζονται, και
2. Το αρχείο papers που περιέχει τις στήλες source_id που είναι το ίδιο με το άνωθεν αρχείο, year που περιέχει την χρονιά δημοσίευσης του άρθρου, title που περιέχει τον τίτλο του, abstract που περιέχει την περίληψη του, και full_text που περιέχει όλο το άρθρο.

Για το κάθε άρθρο που έχουμε επιλέξει θα χρησιμοποιήσουμε index για να εισάγουμε τα έγγραφα στο IndexWriter.

Υλοποίηση αναζητήσεων.

Κατά την λειτουργικότητα 1 της μηχανής αναζήτησης, ο χρήστης θα έχει την δυνατότητα να κάνει αναζήτηση με λέξεις κλειδιά στα πεδία βάσει συγγραφέως, τίτλου, χρονιάς και ινστιτούτου. Αυτό θα υλοποιηθεί με αντεστραμμένα ευρετήρια που θα αφορούν τα πεδία αυτά ξεχωριστά. Για να κάνει ο χρήστης αυτή την αναζήτηση αρχικά θα πρέπει να προσδιορίζει μέσω πεδίων που θα προσφέρουμε με την μηχανή αναζήτησής μας. Δηλαδή θα ορίζει «φίλτρα» αναζήτησης, όπως ποιον τίτλο θα ήθελε να έχει το σύγγραμμά του, την χρονιά δημοσίευσής του, το ινστιτούτο υπό το οποίο δημοσιεύτηκε και τον συγγραφέα του, όπως σε άλλες σύγχρονες εφαρμογές με λειτουργικότητα αναζήτησης. Αφού τα κάνει αυτά, θα

πρέπει να επιλέξει την λειτουργία «Αναζήτηση με βάση λέξη κλειδί» και η αναζήτηση θα πραγματοποιείται.

Κατά την λειτουργικότητα 2 της μηχανής αναζήτησης, ο χρήστης θα ψάχνει λέξεις κλειδιά μέσα από τα πεδία τίτλου, περίληψης και μέσα σε όλο το κείμενο. Μέσω της λειτουργικότητας «Απλή αναζήτηση», ο χρήστης θα εισάγει λέξεις-κλειδιά σε μια μπάρα αναζήτησης. Η μηχανή θα ψάχνει αυτές τις λέξεις κλειδιά μέσα στον τίτλο, το κείμενο και την περίληψη και θα εμφανίζει τα αποτελέσματα που επιθυμεί ο χρήστης και θα κάνει highlight τις λέξεις κλειδιά. Για αυτό θα φτιάξουμε ένα αντεστραμμένο ευρετήριο που θα αφορά τα παραπάνω πεδία τα οποία θα κάνουμε tokenize, δηλαδή θα τα χωρίσουμε σε μικρότερα μέρη και έπειτα μέσω ανάλυσης θα δημιουργήσουμε το ευρετήριο.

Κατά την 3η λειτουργικότητα θα χρησιμοποιήσουμε μια υβριδική αναζήτηση, δηλαδή έναν συνδυασμό από τις 2 παραπάνω αναζητήσεις. Ο χρήστης θα βάζει λέξεις κλειδιά και θα μπορεί να επιλέγει και «φίλτρα» ώστε να βγαίνουν ακόμα πιο συγκεκριμένα αποτελέσματα από τις άλλες 2 αναζητήσεις, απλά θα χρειάζεται ο χρήστης να επιλέγει την επιλογή «Αναζήτηση για προχωρημένους» ή «Σύνθετη Αναζήτηση» (TBA).

Η μηχανή μας θα έχει ιδιότητες που μας παρέχει η Lucene όπως stemming, lemmatization και token normalization τα οποία μας χρειάζονται λόγω της πληθώρας πληροφορίας των εγγράφων. Εφόσον έχουμε άρθρα, θέλουμε να μην χωρίζει τις λέξεις πχ. Play, plays ως 2 αποτελέσματα ούτε τους όρους U.S.A. και USA ως διαφορετικούς.

Διεπαφή.

Σαν πρώτη ιδέα σκεφτήκαμε να χρησιμοποιήσουμε παράθυρα. Αρχικά θα εμφανίζεται ένα παράθυρο όπου ο χρήστης θα επιλέγει είτε αναζήτηση μέσω έτοιμων φίλτρων είτε να πληκτρολογεί μόνος του τι επιθυμεί να ψάξει. Αφού επιλέξει θα πηγαίνει σε άλλο παράθυρο όπου θα βλέπει τα αποτελέσματα της αναζήτησης του. Εκεί θα μπορεί να επιλέξει εκείνο που επιθυμεί και να μεταφερθεί σε ένα τελευταίο παράθυρο, όπου και θα μπορεί τελικά να διαβάσει το άρθρο που επέλεξε. Θα εμφανίζεται τίτλος, σύντομη περίληψη και αφού κάνει κλικ πάνω στον τίτλο, ολόκληρο το άρθρο.

Διάταξη.

Για τις λειτουργικότητες της μηχανής αναζήτησης μας, ο βασικός τρόπος διάταξης θα είναι με βάση τη συχνότητα εμφάνισης των όρων που αναζητά ο χρήστης. Αυτό θα επιτευχθεί μέσω υπολογισμού της συχνότητας όρου και αντίστροφης συχνότητας εγγράφου (tf, idf), καθώς και του βαθμού εγγράφου και ερώτησης

(scoring). Επίσης σκεφτόμαστε να εμφανίζονται πρώτα τα άρθρα που έχουν δημοσιευθεί πιο πρόσφατα.