Πέτρος Κοντός ΑΜ: 2898

Παναγιώτης Χατζηλαζάρου ΑΜ:2856

# Εργασία: Μηχανή αναζήτησης άρθρων σχετικών με τον COVID-19

**Περιγραφή συλλογής**

Για τη συλλογή των άρθρων χρησιμοποιήσαμε την έτοιμη συλλογή από το Kaggle *https://www.kaggle.com/allen-institute-for-ai/CORD-19-research-challenge* την οποία επεξεργαστήκαμε με σκοπό να κρατήσουμε κάποια άρθρα από αυτήν. Πιο συγκεκριμένα από το αρχείο metadata.csv το οποίο έχει περίπου 500 χιλιάδες άρθρα, αρχικά επιλέξαμε αυτά που στον τίτλο περιείχαν τον όρο coronavirus και έχουν δημοσιευτεί το 2021. Από αυτά επιλέξαμε να κρατήσουμε όσα αρχεία εμφανίζονταν σε .json μορφή. Τέλος αποθηκεύσαμε τα αρχεία σε έναν φάκελο corpus τα οποία είναι σε σύνολο 678.

**Περιγραφή μηχανής αναζήτησης**

Στόχος της μηχανής αναζήτησης είναι ο χρήστης να μπορεί να κάνει ερωτήσεις και να του επιστρέφονται τα πιο συναφή, με βάση την ερώτηση, άρθρα. Είναι σημαντικό να μπορεί να επιλέγει διαφορετικούς τρόπους αναζήτησης ώστε να του επιστρέφετε το κατάλληλο αποτέλεσμα από τη μηχανή.

Για την προ επεξεργασία των άρθρων θα χρησιμοποιήσουμε τον StandardAnalyzer της Lucene ο οποίος μπορεί να αναγνωρίζει emails και URLs, όπως επίσης και να αφαιρεί stop words και να μετατρέπει τα tokens σε lowercase.

Η πρώτη μας σκέψη για τα fields είναι να χρησιμοποιήσουμε τον τίτλο του άρθρου, τους συγγραφείς, την περίληψη του κειμένου (abstract), το βασικό κείμενο του άρθρου(body text) και ίσως τα sections στα οποία χωρίζεται το εκάστοτε άρθρο. Όσον αφορά τους τύπους των πεδίων, ο τίτλος, η περίληψη και οι αρθρογράφοι θα οριστούν ως analyzed και stored, όπως επίσης το βασικό κείμενο και τα sections θα είναι analyzed αλλά όχι stored. Στη συνέχεια θα φτιάξουμε ένα ευρετήριο με τη βοήθεια του IndexWriter που μας παρέχει η Lucene το οποίο  το οποίο θα αποθηκεύσουμε στον δίσκο.

Όσον αφορά την αναζήτηση που παρέχει η μηχανή, ο χρήστης θα μπορεί να αναζητά με βάση κάποιο από τα πεδία που αναφέραμε παραπάνω είτε και με τον συνδυασμό τους. Με την βοήθεια του QueryParser θα γίνεται η  κατάλληλη ανάλυση στην ερώτηση που έκανε ο χρήστης, όπως έγινε και στα έγγραφα, και θα δίνει τα αποτελέσματα στο Query. Με την κατάλληλη σύνταξη του QueryParser θα μπορεί να υποστηρίζονται διάφοροι τύποι ερωτημάτων , όπως για παράδειγμα αναζήτηση σε κάποιο πεδίο με όρο ή με φράση αλλά και την κατάλληλη σύνταξη για να υποστηρίζεται η αναζήτηση σε κάποιο πεδίο, με παραπάνω από έναν ορούς.

Για την παρουσίαση των αποτελεσμάτων σε διάταξη με βάση τη συνάφεια τους με το εκάστοτε ερώτημα σκεφτήκαμε να χρησιμοποιήσουμε τις μεθόδους του indexSearcher **TopDocs search()** η

οποία επιστρέφει τα πιο συναφή έγγραφα με βάση το ερώτημα και επίσης κάποια μέθοδο να μας γυρνάει τα αποτελέσματα με βάση κάποιο συγκεκριμένο πεδίο**.** Επίσης, θα χρησιμοποιήσουμε και έναν Highlighter της Lucene ώστε να επισημανθούν οι φράσεις ή οροί που αναζητήθηκαν. Τέλος, για τη διεπαφή του χρήστη με την μηχανή αναζήτησης θα χρησιμοποιήσουμε κάποιο GUI ώστε να μπορεί ο χρήστης να γράψει το ερώτημα του όπως επίσης και να επιλέξει σε ποια πεδία θέλει να γίνει η αναζήτηση.