Ανάκτηση Πληροφορίας 2019-2020
Εργασία: Σχεδιασμός και Υλοποίηση Μηχανής Αναζήτησης

ΠΕΡΙΓΡΑΦΗ

Στην εργασία αυτή θα ασχοληθούμε με το σχεδιασμό και υλοποίηση μίας απλής μηχανής αναζήτησης, η οποία όμως θα έχει όλη τη βασική λειτουργικότητα μίας μηχανής αναζήτησης μεγάλης κλίμακας. Η εργασία λαμβάνει το 40% του τελικού βαθμού (40% η τελική εξέταση και 20% η πρόοδος) και πρέπει να εκπονηθεί σε ομάδες των δύο ατόμων.

Η μηχανή αναζήτησης χονδρικά πρέπει να έχει τα εξής υποσυστήματα:

Crawler: αποτελεί το υποσύστημα που είναι υπεύθυνο για τη συλλογή των δεδομένων από το web. Ο crawler εκτελείται σαν ξεχωριστή process στο background και μαζεύει κείμενο από τις ιστοσελίδες. Στον crawler δίνουμε παράμετρο το URL αφετηρία και επίσης έναν ακέραιο αριθμό που δηλώνει πόσες σελίδες θέλουμε να κάνουμε crawling. Εάν ξεκινήσουμε πάλι τον crawler τα δεδομένα που είχαμε συλλέξει σε προηγούμενη εκτέλεση διατηρούνται εκτός και αν με μία παράμετρο δηλώσουμε ότι θέλουμε να ξεκινήσουμε από την αρχή. Επίσης, μία άλλη παράμετρος στον crawler θα μπορούσε να είναι το πλήθος των threads που θα χρησιμοποιηθούν. Προφανώς, μπορείτε να παραμετροποιήσετε τον crawler με επιπλέον παραμέτρους ανάλογα με τις ανάγκες σας (π.χ. αν θα εκτελέσει BFS, DFS ή κάποιον υβριδικό αλγόριθμο graph search). Ακολουθεί παράδειγμα εκτέλεσης του crawler ο οποίος ξεκινά από τη σελίδα `http://mypage.gr` θα συγκεντρώσει τα στοιχεία από 200 σελίδες, θα διατηρήσει τα προηγούμενα δεδομένα και θα λειτουργήσει με 8 threads:

```
myCrawler http://mypage.gr 200 1 8
```

Indexer: είναι το υποσύστημα που υλοποιεί τον αντεστραμμένο κατάλογο. Θα πρέπει να αποθηκεύετε όλες τις απαραίτητες πληροφορίες ώστε να μπορείτε να υπολογίζετε την ομοιότητα cosine με TF-IDF διανύσματα. Το λεξικό του καταλόγου αποτελείται από όλα τα μοναδικά tokens που συναντούμε μέσα στο κείμενο των σελίδων που κάνουμε crawling. Ο Indexer παίρνει τις πληροφορίες από τον Crawler.

Query Processor: είναι το υποσύστημα που είναι υπεύθυνο για την επεξεργασία ενός ερωτήματος. Ο χρήστης θα πρέπει να μπορεί να δώσει ένα ερώτημα και να ζητήσει τα top-k έγγραφα (webpages) που έχουν τη μεγαλύτερη ομοιότητα με το ερώτημα του χρήστη. Το ερώτημα διατυπώνεται χρησιμοποιώντας μία απλή html σελίδα. Τα αποτελέσματα εμφανίζονται στο χρήστη με φθίνουσα διάταξη ως προς το cosine similarity score, και περιλαμβάνουν τον τίτλο της σελίδας και το URL της σελίδας καθώς και όποια άλλη πληροφορία θέλετε να εμφανίσετε.

Όλα τα υποσυστήματα της μηχανής σας θα πρέπει να μπορούν να εκμεταλλεύονται τον παραλληλισμό και επομένως να είναι σε θέση να χρησιμοποιούν πολλά νήματα για την εκτέλεση της εργασίας.

ΠΑΡΑΔΟΤΕΑ

Η εργασία θα πρέπει να ανεβεί στο elearning στο τέλος Ιανουαρίου 2020 (η ακριβής ημερομηνία θα γίνει γνωστή αργότερα). Θα πρέπει να παραδώσετε αναφορά με τη λύση σας και τον πηγαίο κώδικα της λύσης σας. Μπορείτε να χρησιμοποιήσετε διαφορετικές γλώσσες προγραμματισμού για τα διαφορετικά υποσυστήματα. Επιτρέπεται η χρήση εξωτερικών εργαλείων μόνο για το κομμάτι του preprocessing (π.χ. καθαρισμός, διαγραφή των HTML tags κλπ). Όλα τα υπόλοιπα θα πρέπει να υλοποιηθούν από εσάς.

Καλή επιτυχία!