Μηχανή αναζήτησης άρθρων σχετικών με τον COVID-19 (NOODLE)

Αγαμέμνων Κυριαζής 4400 Χρήστος Περγαμηνέλης 4474

copyright free image

Περιεχόμενα

1. Σύντομη περιγραφή της συλλογής και του τρόπου που επιλέξαμε τα δεδομένα.

1.1. Συλλογή δεδομένων και επεξεργασία τους.

Για τα δεδομένα χρησιμοποιήσαμε τον σύνδεσμο που μας δόθηκε στις διαφάνειες των διαλέξεων και 'κατεβάσαμε' την συλλογή από την ιστοσελίδα (βρείτε τον σύνδεσμο εδώ). Ο συγκεκριμένος φάκελος περιέχει ένα .csv αρχείο με περισσότερο από 350.000 άρθρα σχετικά με τον ιό covid19.

Στη συνέχεια γράψαμε ένα script όπου διαβάζει ένα ορισμένο πλήθος γραμμών του .csv αρχείου και για κάθε γραμμή δημιουργεί ένα ξεχωριστό αρχείο απλού κειμένου (.txt) και την γράφει σε αυτό με το format που θα δούμε παρακάτω. Ο λόγος που επιλέξαμε να παραδοθούν τα αρχεία σε μορφή .txt είναι επειδή η βιβλιοθήκη Lucene δέχεται τέτοια αρχεία.

1.2. Περιγραφή data_extractor.py.

Αρχικά το πρόγραμμα ζητάει από τον χειριστή το πλήρες μονοπάτι του φακέλου που θέλει να τοποθετηθούν τα αρχεία κειμένου όπου πρόκειται να παραχθούν. Επιπλέον περιμένει ως είσοδο στο τερματικό το πλήθος των γραμμών (δηλαδή πόσα αρχεία θα δημιουργήσει) που θα διαβάσει από το .csv αρχείο. Το όνομα κάθε αρχείου κειμένου προκύπτει από την γραμμή που αντιστοιχεί στο csv αρχείο, π.χ. το αρχείο 34.txt προέρχεται από την γραμμή 34 του csv αρχείου. Αφού διαβάσει μια γραμμή από το csv, την 'σπάει' σε στήλες και τις γράφει (αυτές που μας ενδιαφέρουν να αποθηκεύσουμε) στο αρχείο κειμένου χωρισμένες με χαρακτήρα αλλαγής γραμμής ('\n'). Τέλος στα πεδία που μας ενδιαφέρουν είναι και ο αρθρογράφος, όμως κάποιες εγγραφές δεν έχουν συγγραφέα. Σε αυτή την περίπτωση το πρόγραμμα αναλαμβάνει να τοποθετήσει σε αυτό το πεδίο την φράση 'Anonymous Author'.

1.3 Οδηγίες χρήσης του προγράμματος.

Απαραίτητη προϋπόθεση για την ορθή λειτουργεία του προγράμματος είναι το data_extractor.py και το .csv αρχείο να βρίσκονται στον ίδιο κατάλογο, διαφορετικά το πρόγραμμα δεν θα μπορεί να 'δει' το αρχείο με τα δεδομένα και θα προκύψει σφάλμα. Στη συνέχεια απαιτείται από το πρόγραμμα ένας ακέραιος που αντιστοιχεί στο πλήθος γραμμών που θα αναγνωστεί από το csv αρχείο, ξανά αν δεν δοθεί ακέραιος θα προκύψει σφάλμα.

1.4 Περιγραφή δομής των αρχείων δεδομένων.

Τα αρχεία κειμένου έχουν, για την συγκεκριμένη παράδοση (v1.0.) της εργασίας, το εξής format:

Τίτλος Άρθρου: …
Αρθρογράφος: …
Κυρίως Κείμενο: …

Εικόνα 1

Σε επόμενες εκδόσεις της εργασίας σκοπεύουμε να εισάγουμε και το πεδίο της ημερομηνίας. Η μέχρι στιγμής υλοποίηση της εφαρμογής μας απαιτεί το συγκεκριμένο format να παραμείνει αναλλοίωτο για τεχνικούς λόγους.

2. Συνοπτική περιγραφή του σχεδιασμού του συστήματος.

2.1. Γενική περιγραφή: Στόχος και λειτουργικότητα.

Βασικός στόχος της εφαρμογής μας είναι η ανάκτηση πληροφορίας από τα διάφορα έγγραφα που βρίσκονται μέσα στη βάση δεδομένων μας με τον αποτελεσματικότερο και τον πιο φιλικό προς τον χρήστη τρόπο. Για την εφαρμογή μας χρησιμοποιήθηκε η βιβλιοθήκη Apache Lucene 8.8.1. που ήταν η νεότερη έκδοση της βιβλιοθήκης όταν ξεκινήσαμε να γράφουμε τον κώδικα. Η υλοποίηση της εφαρμογής έγινε στην γλώσσα Java1.8. λόγω του μεγάλου πλήθους πηγών σε σχέση με την PyLucene που χρησιμοποιεί την γλώσσα Python.

Η ροή του προγράμματος που έχουμε υλοποιήσει **μέχρι στιγμής** είναι η εξής:

- Ο χειριστής τρέχει το πρόγραμμα και εμφανίζεται ένα παράθυρο διαλόγου (G.U.I.) το όποιο θα του δίνει τις δυνατότητες μέσω κουμπιών είτε να δημιουργήσει από την αρχή την βάση δεδομένων, της οποίας τα δεδομένα προήλθαν από το data_extractor.py και στην οποία θα γίνει η αναζήτηση, είτε να κάνει αναζήτηση με σε μια ήδη υπάρχουσα δομή.
- Συνεχίζοντας ο χειριστής μπορεί να πραγματοποιήσει αναζήτηση πληκτρολογώντας την λέξη, ή την φράση, που τον ενδιαφέρει στο πεδίο κειμένου που δίνεται στην κορυφή του παραθύρου και να οριστικοποιήσει το αίτημα του πατώντας το κουμπί 'Search' στα δεξιά του text – box. Η αναζήτηση μέχρι στιγμής γίνεται μόνο βάσει ολόκληρου του περιεχομένου του αρχείου χωρίς να λαμβάνει υπόψιν του ξεχωριστά πεδία.
- Αφού η εφαρμογή λάβει ένα αίτημα αναζήτησης θα εμφανίσει τα αποτελέσματα στο scrollable – text area του παραθύρου.

Επεκτάσεις που υπολογίζουμε να προσθέσουμε μέχρι την επόμενη παράδοση αποτελούν:

1. Την προσθήκη και αξιοποίηση κατά την αναζήτηση πεδίου ημερομηνίας με κατάλληλη επέκταση του data_extractor.py και της εφαρμογής μας.
2. Την αναζήτηση χρησιμοποιώντας διαφορετικές παραμέτρους (πεδία) κατά την αναζήτηση, όπως αρθρογράφο, τίτλο και ημερομηνία.
3. Μόνιμη αποθήκευση ιστορικού αναζήτησης εσωτερικά της εφαρμογής με χρήση log files και δυνατότητα διαγραφής και εμφάνισης του.
4. Αξιοποίηση του ιστορικού για καλύτερες προτάσεις κατά την αναζήτηση.
5. Δυνατότητα επιλογής εμφάνισης αποτελεσμάτων αναζήτησης με μορφή 'πιο πρόσφατα πρώτα' και 'σχετικότερα πρώτα'.

2.2. Ανάλυση κειμένου και κατασκευή ευρετηρίου.

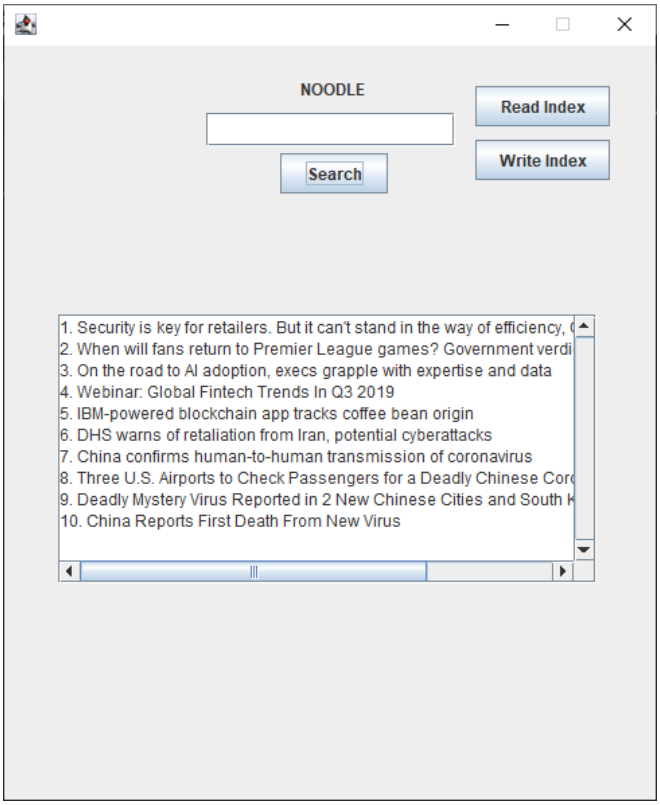
Για την ανάλυση των άρθρων χρησιμοποιήθηκε ο Standard Analyzer της Lucene. Η επεξεργασία που γίνεται από τον συγκεκριμένο Analyzer στο κείμενο αφαιρεί άρθρα και μη κωδικούς χαρακτήρες που δεν χρησιμεύουν στην αναζήτηση. Θεωρείται ο Analyzer με την μεγαλύτερη λειτουργικότητα και για αυτό τον επιλέξαμε. Στη συνέχεια για το indexing των αρχείων προσθέσαμε ως πεδία τον τίτλο του εγγράφου που αντιστοιχεί στην πρώτη γραμμή, τον αρθρογράφο που αντιστοιχεί στην δεύτερη γραμμή και ένα FileReader αντικείμενο για όλα τα περιεχόμενα του εγγράφου (τίτλος, αρθρογράφος και κυρίως κείμενο) μαζί. Ως επιπλέον επέκταση, όπως προαναφέρθηκε, σκοπεύουμε να προσθέσουμε το πεδίο της ημερομηνίας.

2.3. Αναζήτηση.

Όταν ο χειριστής ζητήσει να πραγματοποιήσει αναζήτηση στη βάση που έχει δημιουργηθεί προηγουμένως, το frontend θα μεταβιβάσει το αίτημα στο backend. Η μηχανή θα δεχθεί ως όρισμα ένα αλφαριθμητικό βάσει του οποίου θα δημιουργήσει ένα Query. Από αυτό το Query θα βρει τα δέκα πιο συναφή έγγραφα και θα διαβάσει τους τίτλους τους. Τέλος θα συμπτύξει τους τίτλους σε ένα ενιαίο αλφαριθμητικό και θα το επιστρέψει στο frontend για να το προβάλει στο scrollable text – box που βρίσκεται στο κάτω μέρος του γραφικού περιβάλλοντος. Στην παρούσα παράδοση της εργασίας η είσοδος από τον χειριστή δεν φιλτράρεται από κάποιο analyzer το οποίο θα διαγράψει λέξεις που δεν μας ενδιαφέρουν κατά την αναζήτηση, π.χ. “an”, “a”, “the”. Αυτή η λειτουργία θα υλοποιηθεί σε επόμενη παράδοση.

2.4. Παρουσίαση αποτελεσμάτων.

Το γραφικό περιβάλλον απεικόνισης αποτελεσμάτων στο τρέχον στάδιο της εργασίας φαίνεται στην εικόνα 2



Εικόνα 2

Επιπλέον προσθήκες στο γραφικό περιβάλλον για την παρουσίαση των αποτελεσμάτων, όπως επιλογή κατάταξης με βάση την ημερομηνία και βελτιώσεις στα γραφικά, θα υλοποιηθούν στην επόμενη παράδοση.