Πανεπιστήμιο Ιωαννίνων
Τμήμα Μηχανικών Ηλεκτρονικών Υπολογιστών και Πληροφορικής

Ανάκτηση Πληροφορίας

2η Φάση

Μέλη ομάδας:

- Γεώργιος Δεληγιώργης, 4662
- Ελευθέριος-Μάριος Μανίκας, 4723

ΑΚΑΔΗΜΑΪΚΟ ΕΤΟΣ 2024
ΠΑΡΑΔΟΣΗ: Τετάρτη 22 Μαΐου

Περιεχόμενα

1.Github link

https://github.com/manikas07/Anaktish-Plhroforias.git

Η παρούσα αναφορά περιγράφει τα βήματα υλοποίησης της 2ης φάσης στην άσκηση για το μάθημα της Ανάκτησης Πληροφορίας.

2. Συλλογή Εγγράφων

Όπως μας υποδεικνύεται από την εκφώνηση κατεβάσαμε τα έγγραφα με τα επιστημονικά κείμενα από το παρακάτω link https://www.kaggle.com/datasets/rowhitswami/nips-papers-1987-2019-updated/data?select=papers.csv. Επιλέξαμε από ένα σύνολο 9680 άρθρων 488 άρθρα για να τα χρησιμοποιήσουμε ως την βάση δεδομένων μας. Το format του αρχείου είναι .csv(comma separated values).

- Στην πρώτη στήλη έχουμε την χρονιά που δημοσιοποιήθηκε το κάθε άρθρο.
- Στην δεύτερη στήλη τον τίτλο του.
- Στην τρίτη στήλη το abstract που είναι μια μικρή περίληψη του άρθρου και μπορεί να παραλείπεται, δεν έχουμε επιλέξει άρθρα που έχουν abstract.
- Στην τέταρτη στήλη είναι όλο το υπόλοιπο κείμενο του άρθρου.

Το αρχείο το επεξεργαστήκαμε αρκετά έτσι ώστε να γίνεται σωστά η κατανομή των πεδίων μέσω της κλάσης LuceneIndex.Αρχικά αφαιρέθηκε το πεδίο source_id διότι δεν θα το χρησιμοποιήσουμε στην υλοποίηση μας. Επίσης αφαιρέθηκαν τα κόμματα, τα εισαγωγικά, και η αλλαγή γραμμής και παραγράφου από το πεδίο full_text έτσι ώστε η μέθοδος indexDocuments να διαβάζει σωστά το αρχείο.

3. Περιγραφή Σχεδιασμού Συστήματος

3.1 Στόχος & Λειτουργικότητα

Ο στόχος για την συγκεκριμένη εργασία είναι να δημιουργήσουμε ένα interface στο οποίο ο χρήστης θα μπορεί να θέτει ερωτήματα τα οποία θα σχετίζονται με επιστημονικά άρθρα που είναι αποθηκευμένα στην Βάση Δεδομένων. Αυτό επιτυγχάνεται μέσω μιας γραφικής διεπαφής (GUI) και το σύστημα μας θα τις διαχειρίζεται με την βοήθεια της βιβλιοθήκης ανοιχτού κώδικα Lucene και θα τα επεξεργάζεται αναλόγως των οδηγιών της εκφώνησης.

3.2 Ανάλυση κειμένου και κατασκευή ευρετηρίου

Για την ανάλυση του .csv αρχείου επιλέξαμε τον StandardAnalyzer της Lucene διότι μετατρέπει όλους τους χαρακτήρες σε lower case και απομακρύνει τις περιττές λέξεις(π.χ. the, and, or...).Αρχικά δημιουργούμε 2 μεταβλητές τύπου String έτσι ώστε να αποθηκεύσουμε σε αυτά τα path τα αρχεία ευρετηρίου της Lucene μετά την επεξεργασία του csv αρχείου και το path για να βρει το πρόγραμμα μας το αρχείο papers.csv. Η indexDocuments χρησιμοποιείται για να διαβαστεί το αρχείο csv και να μετατραπούν τα δεδομένα του αρχείου σε ευρετήριο της lucene. Αν η γραμμή έχει τέσσερα πεδία, δημιουργείται ένα νέο έγγραφο (Document). Στο έγγραφο προστίθενται πεδία με τις αντίστοιχες τιμές:

year: το πρώτο πεδίο

title: το δεύτερο πεδίο

abstract: το τρίτο πεδίο

full_text: το τέταρτο πεδίο

Κάθε πεδίο προστίθεται ως TextField με την επιλογή Field.Store.YES, που σημαίνει ότι η τιμή του πεδίου θα αποθηκευτεί στο ευρετήριο. Τέλος το document περνάει στον IndexWriter ο οποίος αποθηκεύει τα έγγραφα στο ευρετήριο. Έπειτα στην μεθοδο search() χρησιμοποιούμε το API org.apache.lucene.search για να ανοίξει το αρχείο του ευρετηρίου, να δημιουργήσει έναν IndexReader για το διάβασμα του ευρετηρίου και έναν IndexSearcher για την αναζήτηση εγγράφων μέσα στο ευρετήριο. Στην συγκεκριμένη μέθοδο η αναζήτηση γίνεται με βάση το year. Χρησιμοποιούμε τον StandardAnalyzer για την ανάλυση της συμβολοσειράς. Επίσης ο QueryParser χρησιμοποιεί τον analyser για να μετατρέψει την συμβολοσειρά σε ένα αντικείμενο query. Τέλος ο searcher επιστρέφει τα κορυφαία έγγραφα TopDocs και εκτυπώνονται τα αποτελέσματα.

3.3 Αναζήτηση

Αρχικά, η αναζήτηση γίνεται μέσω ενός γραφικού περιβάλλοντος στο οποίο ο χρήστης εισάγει την εκάστοτε ερώτηση που επιθυμεί και πρέπει να επιλέξει αν θέλει αναζήτηση με βάση τον τίτλο ή με την χρονιά ή με λέξη κλειδί στο πεδίο του abstract. Αν δεν κάνει κάποια επιλογή τότε η αναζήτηση γίνεται με λέξη κλειδί στο πεδίο του full_text. Επίσης η μέθοδος αναζήτησης μας επιτρέπει να ψάξουμε και τίτλο με μια λέξη κλειδί. Να σημειωθεί ότι δεν επιτρέπεται η αναζήτηση με πάνω από ένα check box επιλεγμένο.

3.4 Παρουσίαση Αποτελεσμάτων

Αρχικά μετά την εισαγωγή του ερωτήματος και την επιλογή του checkbox ο χρήστης πατάει το κουμπί search και έτσι εμφανίζονται στην result area ταξινομημένα κατά φθίνουσα σειρά συνάφειας ανά 10. Δίνεται η δυνατότητα στο χρήστη να πατήσει το κουμπί more με το οποίο θα δει τις παρακάτω σελίδες με τις καταχωρίσεις. Οι λέξεις που ταιριάζουν στην αναζήτηση του χρήστη εμφανίζονται στα αποτελέσματα πρασινισμένες. Αριστερά είναι το πεδίο του search history το οποίο ανανεώνεται αυτόματα μετά από κάθε αναζήτηση. Μέσω των μεθόδων που έχουμε αποθηκεύεται σε ένα αρχείο κάθε αναζήτηση η οποία θα αξιοποιηθεί για να προτείνει σε άλλον χρήστη μελλοντικά ερωτήματα τα οποία εμφανίζονται στην κονσόλα όταν τρέχουμε την εφαρμογή και όχι σε κάποιο jpanel. Τέλος δίνεται η δυνατότητα μέσω στον χρήστη μέσω του κουμπιού Sort by Year να ταξινομήσει τα αποτελέσματα που βλέπει στο παράθυρο με φθίνουσα χρονολογική σειρά.

Για να επιτευχθούν οι παραπάνω λειτουργίες έχουμε φτιάξει μια μέθοδο για κάθε λειτουργία που παρέχουμε στον χρήστη.

Πιο συγκεκριμένα για να κάνουμε highlight τα δεδομένα χρησιμοποιούμε την μέθοδο highlightText() η οποία αξιοποιεί την βιβλιοθήκη javax.swing.text.Highlighter.

Για να εμφανίσουμε τα αποτελέσματα στο jpanel χρησιμοποιούμε την μέθοδο displayResults() η οποία μας εμφανίζει τον τίτλο, την χρονιά δημοσίευσης και ένα μικρό αρχικό κομμάτι του full_text κάθε φορά που την καλούμε.

Η μέθοδος performSearch() ουσιαστικά είναι υπεύθυνη για την αναζήτηση του ερωτήματος στο ευρετήριο της lucene που ήδη έχει δημιουργηθεί και επιστρέφει TopDocs results τα οποία είναι τα αποτελέσματα της ερώτησης μας ,δηλαδή τα papers που έχουν μεγαλύτερη συνάφεια με το ερώτημα μας.Τέλος η κλάση SearchHistoryItem είναι υπεύθυνη για την αποθήκευση της ερώτησης στο ιστορικό και την επιστροφή τον προτεινομένων μελλοντικών ερωτήσεων και συνδέεται με την κλάση LuceneGui η οποία δημιουργήθηκε και με την βοήθεια του window builder plugin του eclipse για την δημιουργία ενός γραφικού περιβάλλοντος. Το όνομα του παραθύρου που χρησιμοποιείται ως διεπαφή είναι «Search Scientific Articles» . Η ανανέωση του

παραθύρου μετά από κάθε search γίνεται αυτόματα δίνοντας μας την δυνατότητα να δίνουμε στο πρόγραμμα πολλές ερωτήσεις χωρίς να χρειάζεται κάθε φορά να τρέχουμε από την αρχή την εφαρμογή μας.