Χρήστος Μιχίδης 3030

Ζήσης-Νικόλας Γεωργάκης 2950

Περίληψη:

Αρχικά θα περιγράψουμε τις βασικές λειτουργίες του συστήματος μας εδώ, και στη συνέχεια θα δώσουμε μια ποιο εκτενείς εξήγηση για κάθε μία λειτουργία.

Το σύστημα μας ως προς τις βασικές λειτουργίες, εκτελεί μια μορφοποίηση του αρχείου εισόδου, κάνει indexing και δίνει τα fields ως προς τα οποία γίνεται η αναζήτηση, και εκτελεί το query. Βασικό πρόβλημα/δυσλειτουργία είναι ότι στην εμφάνιση των αποτελεσμάτων εμφανίζει μόνο το πρώτο αποτέλεσμα. Αν και δίνει hits μετρώντας πέρα από την πρώτη επιτυχία, όπου δείχνει και τον αριθμό τον hits, δεν βρήκαμε τρόπο να λύσουμε αυτό το βασικό πρόβλημα.

Ως προς τις ίδιες τις λειτουργίες, επιτρέπουμε στο χρήστη να κάνει αναζήτηση με βάση ένα πεδίο. Ωστόσο, αν ο χρήστης έχει τη γνώση της σύνταξης query στη lucene μπορεί να κάνει πιο περίπλοκες αναζητήσεις. Επίσης υπάρχουν 2 πεδία, τα date_added και το duration, τα οποία δεν τα συμπεριλαμβάνουμε στην αναζήτηση. Θεωρήσαμε πως τα 2 αυτά πεδία κανείς δεν τα χρησιμοποιεί για αναζήτηση, και απλά θεωρήσαμε λογικό να τα παραβλέψουμε. Αν χρειαστεί, είναι πολύ εύκολο το σύστημα να προσαρμοστεί ώστε να γίνονται και οι αναζητήσεις σε αυτά τα πεδία.

Σημαντικό ακόμη είναι πως για το σύστημα μας θεωρούμε έναν χρήστη-γνώστη ο οποίος δίνει εισόδους οι οποίες είναι λογικές με βάση τα ζητούμενα. Αν ο χρήστης αποφασίσει να δώσει «παράλογες» εισόδους το σύστημα μας δεν έχει τρόπους αντιμετώπισης, με εξαίρεση σε κάποια σημεία που υπάρχει default είσοδος.

Πέρα από τα θέματα κωδικοποίησης, δίνουμε στο χρήστη δυνατότητα να κάνει αρκετές αναζητήσεις κάνοντας μία φορά indexing.

Από εδώ και κάτω υπάρχει η εκτενείς εξήγηση και λεπτομέρειες του συστήματός μας.

Αρχεία αποστολής

Τα αρχεία που θα σταλθούν είναι σε μορφή eclipse project. Το αρχείο εισόδου nextlix_titles βρίσκετε στο φάκελο inputFiles ενώ τα αρχεία που γίνονται indexed αποθηκεύονται στο φάκελο indexedFiles. Το eclipse project περιέχει τους δύο αυτούς φακέλους ώστε να κάνουμε φόρτωση τον αρχείων χωρίς να δίνετε κάποιο path του υπολογιστή, και επομένως να δουλεύει σε διαφορετικούς υπολογιστές χωρίς καμία αλλαγη στον κώδικα. Μαζί με τα αρχεία υπάρχει και βίντεο που εξηγεί περεταίρω κάποια πράγματα.

Μορφοποίηση δεδομένων

Ξεκινάμε με την μορφοποίηση που κάναμε στης είσοδο μας. Ως είσοδο έχουμε ένα αρχείο, το netflix_titles.csv το οποίο περιέχει κάπου στις 9000 εγγραφές. Αποφασίσαμε να περιλάβουμε όλες τις εγγραφές. Ως document, όπως είχαμε σκεφτεί στην αρχή θεωρήσαμε κάθε γραμμή του αρχείου, δηλαδή κάθε εγγραφή ταινίας. Παραλείψαμε την πρώτη γραμμή που απλά ονομάτιζε τα πεδία.

Στα θέματα coding. Αρχικά για την εισαγωγή του αρχείου χρησιμοποιήσαμε την κλάση scanner. Το αρχείο το στείλαμε μαζί με το eclipse project στο φάκελο inputFiles, και το χειριζόμαστε με τρόπο ώστε να μην χρειάζεται αλλαγή ή είσοδος άλλου path για να δουλέψει σε άλλο υπολογιστή.

Όταν δεχθήκαμε το αρχείο, χρησιμοποιώντας μία for, στέλνουμε για indexing μία μία τις γραμμές. Εδώ υπάρχει το πρόβλημα στο πεδίο listed_in που είναι το μόνο που χωρίζεται με αυτάκια («») και περιέχει μέσα δεδομένα που χωρίζονται με κόμμα (,). Το πρόβλημα είναι πως με την εντολή split

χωρίσαμε τα πεδία μέχρι εκεί, και στην συγκεκριμένη περίπτωση απλά χωρίζει το listed_in στις δύο πρώτες επιλογές του πεδίου και το πεδίο description δεν περιλαμβάνεται καθόλου. Τελικά οι αναζητήσεις στα πεδία listed_in και description εμφανίζουν λάθη (σε σπάνιες περιπτώσεις έχουμε σωστό αποτέλεσμα).

Indexing

Το indexing σχεδόν υλοποιείται ταυτόχρονα με την μορφοποίηση των δεδομένων. Για κάθε γραμμή στην for που διαβάζουμε τις γραμμές του αρχείου, στέλνουμε στην μέθοδο addDoc τα πεδία σε έναν πίνακα, και η μέθοδος τα κάνει addDocument στον writer με όνομα w. Επειδή τα indexed αρχεία επαναχρησιμοποιούνται, αντί για addDocument κάνουμε updateDocument, έτσι ώστε να επαναγράφονται. Χωρίς να ξέρουμε γιατί, η updateDocument εντολή δεν κάνει update άμα κάνουμε καινούριο indexing. Για αυτό κάθε φορά που γίνεται indexing πρέπει με το που κλείσει το πρόγραμμα να διαγράφονται τα αρχεία που δημιουργούνται από το indexing. Αλλιώς εμφανίζονται διπλότυπα(τριπλότυπα κτλπ αναλόγως πόσες φορές τρέξαμε το πρόγραμμα) στο αποτέλεσμα.

Τα αρχεία που γίνονται indexing, τα αποθηκεύουμε σε φάκελο μέσα στο eclipse project, και έτσι δεν χρειάζεται συγκεκριμένο path του υπολογιστή.

Search

Η αναζήτηση υλοποιείτε με το που γίνουν indexed όλα τα δεδομένα. Πρώτη είσοδος ζητάτε στον χρήστη ως προς το ποιο πεδίο να κάνει αναζήτηση. Στην κονσόλα δίνονται ως οδηγίες τα νούμερα τα οποία αντιστοιχούν σε κάθε πεδίο. Η υλοποίηση αυτής της λειτουργίας γίνεται στη μέθοδο findIndex. Σε περίπτωση που ο χρήστης δώσει οποιαδήποτε άλλη είσοδο θεωρούμε ως default τον τίτλο.

Έπειτα ζητάμε από το χρήστη το κλειδί ως προς το οποίο θα γίνει η αναζήτηση. Στο τέλος του κλειδιού που δόθηκε προσθέτουμε το χαρακτήρα «*» χωρίς να έχει δοθεί από το χρήστη, έτσι ώστε να γίνεται αναζήτηση με το κλειδί και να υπάρχουν αποτελέσματα τα οποία το περιλαμβάνουν μερικώς σε κάποια λέξη.

Στη συνέχεια, αφού υλοποιήσαμε τους reader , searcher της lucene, παίρνουμε τα αποτελέσματα σε έναν πίνακα με την κλάση TopDocs και εμφανίζουμε το συνολικό αριθμό των επιτυχών ευρέσεων. Στη συνέχεια, με μία for προσπαθούμε να εμφανίσουμε τα αποτελέσματα ανά 10 ως προς το πεδίο που ζητήθηκε μαζί με τον αριθμό id της ταινίας στην οποία βρέθηκε το κλειδί. Όπως αναφέρθηκε και πριν στην περίληψη, χωρίς να ξέρουμε γιατί και που είναι το λάθος, εμφανίζεται μόνο η πρώτη εύρεση τόσες φορές όσες ζητήθηκε.

Τέλος, αφότου εμφανιστούν τα αποτελέσματα, δίνεται η δυνατότητα στο χρήστη να δώσει άλλο query για αναζήτηση, χωρίς να ξαναγίνει indexing. Αυτό επιτυγχάνεται με μία απλή επανάληψη while, έως ότου ο χρήστης να μην θέλει να κάνει άλλη αναζήτηση.

Όλες οι ερωτήσεις συστήματος στις οποίες δίνει απάντηση ο χρήστης, έγιναν με την χρήση της κλάσης scanner.