ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ
ATHENS UNIVERSITY OF ECONOMICS AND BUSINESS

Βάσεις Δεδομένων-Project

Πέτρος Τσότσι (3180193)
Επαμεινώνδας Ιωάννου (3140059)
Παναγιώτης Κάτσος (3180077)

2019-2020

1 Προεπεξεργασία δεδομένων

1.1 Δημιουργία πινάκων

Αρχικά ξεκινήσαμε με την δημιουργία των πινάκων με την βοήθεια του python script της δεύτερης εργασίας και βασιζόμενοι στα δεδομένα μας ορίσαμε τους κατάλληλους τύπους για το καθε πεδίο. Τα συγκεκριμένα sql scripts που χρησιμοποιήθηκαν υπάρχουν στο αρχείο partA.sql. Ενδεικτικά το sql script που χρησιμοποιήθηκε για τον πίνακα Links:

```
create table ‘‘Links’’(
   movieId int,
   imdbId int,
   tmdbId int
);
```

1.2 Διαγραφή διπλοτύπων

Στην συνέχεια επεξεργαστήκαμε τα δεδομένα σε καθε πίνακα με απώτερο σκοπό την προσθήκη πρωτευόντων κλειδιών. Συγκεκριμένα εντοπίσαμε σε κάθε πίνακα (εκτός του Ratings_Small) τα διπλότυπα και ύστερα προχωρήσαμε στην διαγραφή τους. Η λογική που ακολουθήσαμε για τη διαγραφή των διπλοτύπων, ήταν να βρούμε τις εγγραφές που υπάρχουν πάνω από 1 φορά στους πίνακες και να διαγράψουμε τις έξτρα ώστε να γίνουν μοναδικές. Σε όλους τους πίνακες (πέραν του Ratings_Small) εντοπίστηκαν διπλότυπα. Πιο συγκεκριμένα:

- Στον Credits εντοπίστηκαν και διαγράφηκαν 44 διπλότυπα (45475 εγγραφές πριν τη διαγραφή, 45431 μετά τη διαγραφή)

- Στον Keywords εντοπίστηκαν και διαγράφηκαν 987 διπλότυπα (46419 εγγραφές πριν τη διαγραφή, 45432 μετά τη διαγραφή)
- Στον Links εντοπίστηκαν και διαγράφηκαν 30 διπλότυπα (45843 εγγραφές πριν τη διαγραφή, 45813 εγγραφές μετά τη διαγραφή)
- Τέλος στον Movies_Metadata εντοπίστηκαν 30 διπλότυπα (45463 εγγραφές πριν τη διαγραφή, 45433 εγγραφές μετά τη διαγραφή).

Ενδεικτικά το query που χρησιμοποιήθηκε για διαγραφή διπλοτύπων (παρόμοια χρησιμοποιήθηκαν και για τους υπόλοιπους πίνακες):

```
DELETE FROM ”Credits” as a USING (
      SELECT MIN(ctid) as ctid, id
        FROM ”Credits”
        GROUP BY id HAVING COUNT(*) >1
      )as b
      WHERE a.id = b.id
      AND a.ctid <> b.ctid;
```

Στην ουσία διαγράφονται όλες οι εγγραφές που έχουν count> 1,δηλαδή όσες αποτελούν διπλότυπα, ώσπου να μείνει μία και μοναδική το οποίο εξασφαλίζεται από τη δεύτερη συνθήκη στο WHERE clause.

1.3 Διαγραφή ταινιών

Μετά τη διαγραφή των διπλοτύπων προχωρήσαμε στη διαγραφή ταινιών οι οποίες δεν υπήρχαν στον πίνακα Movies_Metadata αλλά υπήρχαν σε κάποιον από τους υπόλοιπους πίνακες. Οι μόνοι πίνακες για τους οποίους εντοπίστηκε κάτι τέτοιο, ήταν ο Ratings_Small και ο Links. Αναλυτικότερα:

- Στον Ratings_Small εντοπίστηκαν και διαγράφηκαν 55015 εγγραφές που ικανοποιούσαν την παραπάνω συνθήκη(100004 εγγραφές πριν τη διαγραφή των ταινιών, 44989 μετά τη διαγραφή)
- Και στον Links εντοπίστηκαν και διαγράφηκαν 350 εγγραφές (45813 εγγραφές πριν τη διαγραφή των ταινιών, 45433 μετά τη διαγραφή).

Ενδεικτικά το query που χρησιμοποιήθηκε για τη διαγραφή των ταινιών στον Links (παρόμοιο χρησιμοποιήθηκε και για τον Ratings_Small)

```
DELETE FROM ‘‘Links’’ WHERE movieid IN
    (SELECT movieid FROM ‘‘Links’’
    LEFT JOIN ‘‘Movies_Metadata’’
    ON ‘‘Links’’.tmdbid=‘‘Movies_Metadata’’.id
    WHERE ‘‘Movies_Metadata’’.id is NULL )
```

Με το left join και το συγκεκριμένο where clause διαγράφονται οι εγγραφές του Links με movieid που δεν υπάρχει στον πίνακα Movies_Metadata. Στη συγκεκριμένη περίπτωση χρειάστηκε το left join γιατί μόνο έτσι μπορούμε να ανακτήσουμε τις τιμές που μας ενδιαφέρουν και έχουν null value στον Movies_Metadata.

1.4 Πρόσθετες αλλαγές (μετατροπή τύπου σε json)

Μια επιπλέον αλλαγή που έγινε στα δεδομένα ήταν μια μικρή επεξεργασία στο πεδίο genres του πίνακα Movies_Metadata. Έγινε replace το single quote με double quote. Αυτή η αλλαγή έγινε καθώς ο τύπος json απαιτεί double quotes. Το συγκεκριμένο πεδίο χρειάζεται για τα queries του partB και έπρεπε να το μετατρέψουμε σε τύπο json ώστε να έχουμε πιο εύκολη πρόσβαση στα στοιχεία.