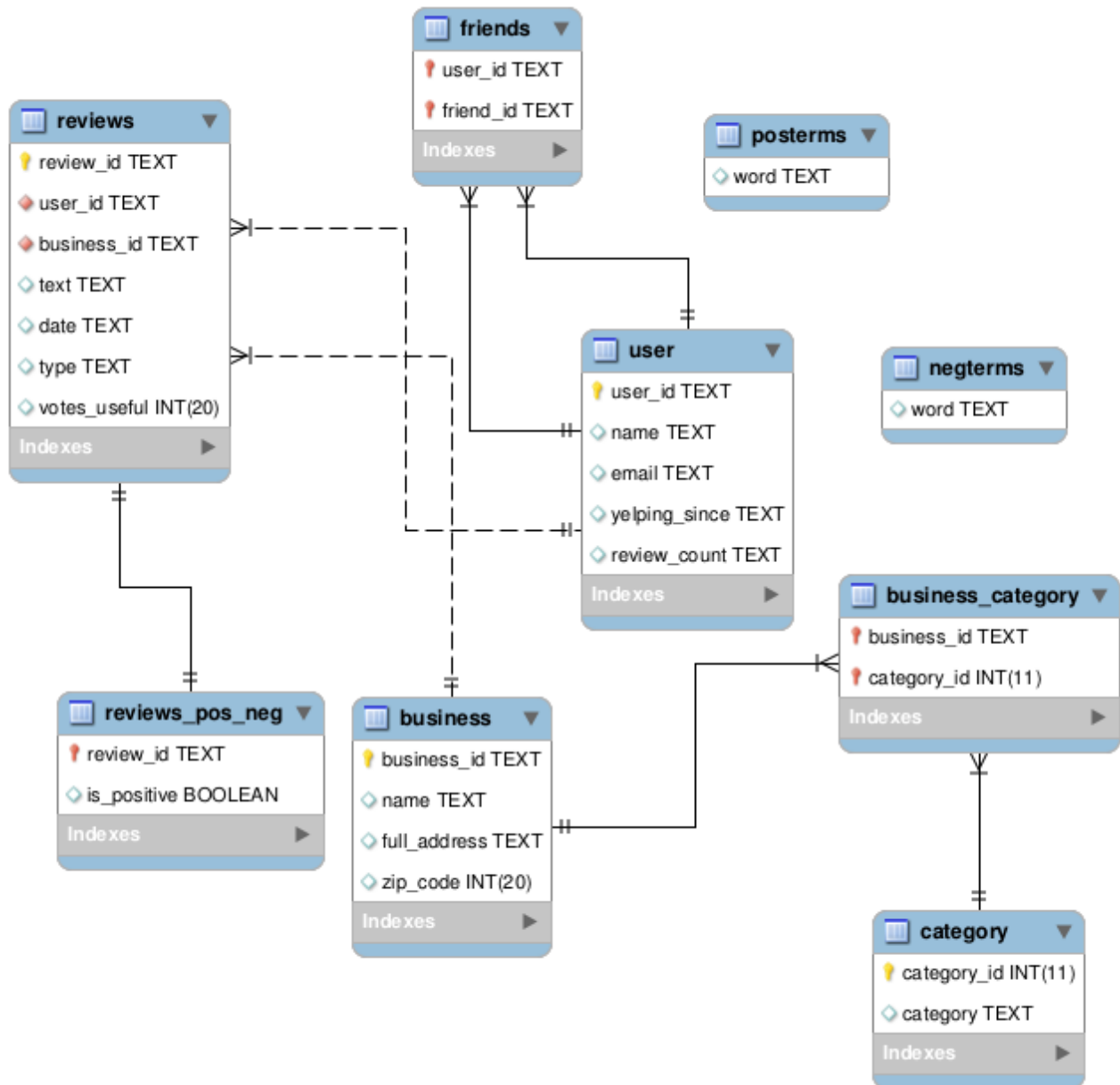


Σχεδίαση και Χρήση Βάσεων Δεδομένων
Εαρινό Εξάμηνο 2017
3η Άσκηση
Χτίζοντας μια εφαρμογή με Python και SQLite (+madIS)
Παράδοση: 2 Ιουνίου 2017, ώρα 23:59

Δεδομένα

Σε αυτή την εργασία, σας δίνεται μια βάση δεδομένων η οποία περιέχει τις αξιολογήσεις/κριτικές των επιχειρήσεων στην ιστοσελίδα: <https://www.yelp.com>. Το σχήμα της βάσης είναι το παρακάτω:



Επίσης σας δίνεται μια εφαρμογή τριών επιπέδων που έχει χτιστεί πάνω από τη βάση. Αποτελείται από τη διεπαφή χρήστη, που είναι web-based, τη λογική της εφαρμογής, που είναι σε python, και ένα σύστημα διαχείρισης βάσεων δεδομένων, που λέγεται madIS και είναι χτισμένο πάνω στην SQLite.

Το παρακάτω είναι η αρχική σελίδα που πρέπει να δείτε στην εφαρμογή:

Classify review (with madis UDFs) Review id: <input type="text"/> <input type="button" value="Search"/>	Classify review (using plain sql) Review id: <input type="text"/> <input type="button" value="Search"/>
Update zip code Business Id: <input type="text"/> Zip code: <input type="text"/> <input type="button" value="Search"/>	Top N businesses per Category Category Id: <input type="text"/> N: <input type="text"/> <input type="button" value="Search"/>
Trace User Influence User Id: <input type="text"/> Depth : <input type="text"/> <input type="button" value="Search"/>	

Ζητούμενο

Στο πλαίσιο της άσκησης, καλείστε να αλλάξετε τη λογική της εφαρμογής και να υλοποιήσετε τις παρακάτω συναρτήσεις σε Python, οι οποίες επικοινωνούν με τη βάση δεδομένων και πραγματοποιούν κάποιες συγκεκριμένες λειτουργίες. Όλες οι συναρτήσεις επιστρέφουν μια λίστα από πλειάδες όπου πάντα η πρώτη πλειάδα είναι η κεφαλίδα με τα ονόματα των πεδίων και οι υπόλοιπες είναι τα αποτελέσματα, για παράδειγμα, [(“Name”, “Id”), (“Jim”, 7), (“Tom”, 13,)]. Μέσα στα ερωτήματα SQL που θα έχει η υλοποίηση, θα χρειαστεί να χρησιμοποιήσετε κάποιες συναρτήσεις ορισμένες από το χρήστη (UserDefinedFunctions) που υπάρχουν υλοποιημένες στο madIS.

Συναρτήσεις

1. **classify_review***: Η συνάρτηση αυτή παίρνει ως όρισμα τον κωδικό μιας αξιολόγησης. Βρίσκει το πλήθος των θετικών ή αρνητικών όρων που υπάρχουν μέσα στο κείμενο των αξιολογήσεων και ταξινομούν τις αξιολογήσεις ως θετικές ή αρνητικές. Για να το βρει αυτό, παίρνει το κείμενο της αξιολόγησης λέξη προς λέξη και βρίσκει τους θετικούς ή αρνητικούς όρους που υπάρχουν μέσα σε αυτό. Οι όροι αποτελούνται από μία ή δύο ή τρεις λέξεις. Το παραπάνω μπορεί να γίνει με ένα ερώτημα SQL χρησιμοποιώντας σε αυτό τον τελεστή textwindow του madIS. Οι απαραίτητες πληροφορίες για τον τελεστή textwindow είναι διαθέσιμες εδώ:

<https://madgik.github.io/madis/row.html?highlight=textwindow#functions.row.text.textwindow>

Επιστρέφει:

- Το όνομα της επιχείρησης.
- Το θετικό ή αρνητικό σχόλιο.

Hint: Για να βγούν σωστά αποτελέσματα στην πρώτη ερώτηση έχει σημασία ο αριθμός των λέξεων στη φράση που έχει βρεθεί. Αν η θετική φράση περιέχει 3 λέξεις τότε ο μετρητής θετικών φράσεων πρέπει να αυξηθεί κατά 3. Επίσης, όταν μια φράση έχει ήδη μετρηθεί, π.χ., “good food”, τότε δεν πρέπει να μετριέται ξεχωριστά ο όρος “good” που είναι substring της αφού έχει ήδη μετρηθεί σαν λέξη της πλήρους φράσης.

2. **classify_review_plain_sql***: Η συνάρτηση αυτή κάνει ό,τι και η παραπάνω χωρίς τη χρήση τελεστών του madIS, χρησιμοποιώντας δηλαδή απλά ερωτήματα SQL και ό,τι υλοποίηση χρειάζεται στην πλευρά της python.
- * Στις συναρτήσεις (1,2) απαγορεύεται να χρησιμοποιήσετε τη σχέση reviews_pos_neg. Η σχέση αυτή περιέχει για κάθε κριτική το αν είναι θετική ή αρνητική (0,1) και μπορείτε να τη χρησιμοποιήσετε μόνο για να ελέγξετε τα αποτελέσματα σας αλλά και σε επόμενα ερωτήματα που είναι απαραίτητη.
3. **update_zip_code**: Η συνάρτηση αυτή παίρνει από το χρήστη ως ορίσματα την ταυτότητα μιας επιχείρησης και ενημερώνει το πεδίο zip_code της διεύθυνσης αυτής με την τιμή που δίνει ο χρήστης. Σε περίπτωση επιτυχίας, επιστρέφει 'ok'. Αντίθετα, αν δεν υπάρχει τέτοια ταυτότητα επιστρέφει 'error'.
4. **selectTopNbusinesses**: Η συνάρτηση αυτή δέχεται ως όρισμα τον κωδικό μιας κατηγορίας και έναν ακέραιο N. Βρίσκει τις N επιχειρήσεις ανά κατηγορία με βάση το πλήθος των θετικών αξιολογήσεων. Επιστρέφει τα εξής:
- Τον κωδικό της επιχείρησης.
 - Το πλήθος των θετικών αξιολογήσεων για την κάθε επιχείρηση.
5. **traceUserInfluence**: Η συνάρτηση αυτή δέχεται ως όρισμα τον κωδικό ενός χρήστη (user_id). Στην συνέχεια, υπολογίζει το μεταβατικό εγκλεισμό (Transitiveclosure) της επιρροής του χρήστη αυτού σε άλλους χρήστες ως προς την επιλογή επιχειρήσεων και επιστρέφει τους κωδικούς των χρηστών που επηρεάζει.

Hint: Πώς επηρεάζεται ένας χρήστης από έναν άλλον; Ένας χρήστης a επηρεάζει ένα χρήστη b όταν είναι φίλοι και έχουν αξιολογήσει την ίδια επιχείρηση με την αξιολόγηση του a να προηγείτε χρονικά του b. Η επιρροή αυτή δημιουργεί ένα γράφο όπου οι κόμβοι του είναι χρήστες και οι κατευθυνόμενες ακμές του αναπαριστούν επιρροές (υπάρχει κατευθυνόμενη ακμή ανάμεσα σε δύο κόμβους a και b αν ο a επηρέασε τον b). Αυτό που θέλουμε είναι να υπολογίσουμε το μεταβατικό εγκλεισμό αυτού του γράφου¹. Το αποτέλεσμα περιέχει τους χρήστες που επηρεάζει ο δεδομένος χρήστης.

Για παράδειγμα, αν ο a έχει επηρεάσει τον b και ο b έχει επηρεάσει τον c, τότε και ο a έχει επηρεάσει τον c (έμμεσα). Δηλαδή, αν $a \rightarrow b$ και $b \rightarrow c$, τότε στο αποτέλεσμα πρέπει να εμφανίζονται τα εξής:

- (b,)
- (c,) // Λόγω μεταβατικού εγκλεισμού

Σημείωση: Υποθέτουμε ότι το σύνολο των δεδομένων είναι μεγάλο και δεν μπορεί να χωρέσει ολόκληρο στη μνήμη. Δεδομένου αυτού, θα πρέπει να προσπελάσετε το υποσύνολο των δεδομένων που χρειάζεται να εμφανιστούν στο αποτέλεσμα (θα πρέπει να εκτελέσετε παραπάνω από ένα ερώτημα στην βάση). Επίσης σας δίνεται σαν όρισμα το βάθος στο οποίο θα ψάξετε. Για παράδειγμα αν το βάθος είναι 1 τότε θα ψάξετε μόνο τους φίλους του συγκεκριμένου χρήστη ενώ αν το βάθος είναι 2 θα ψάξετε μέχρι και τους φίλους των φίλων του χρήστη.

¹http://en.wikipedia.org/wiki/Transitive_closure

Οδηγίες

Η εφαρμογή βρίσκεται στο app.py. Για να την τρέξετε, θα πρέπει να κάνετε τα εξής:

- Προτεινόμενη πλατφόρμα υλοποίησης: Linux-Ubuntu.
- Να εγκαταστήσετε την Python (version 2.7.x) στον υπολογιστή σας – Στα Linux Ubuntu είναι προεγκατεστημένη.
- Να εγκαταστήσετε το madIS ακολουθώντας τις οδηγίες στο <http://madgik.github.io/madis/install.html>. Τεκμηρίωση για το madIS μπορείτε να βρείτε στο σύνδεσμο <http://madgik.github.io/madis/>. Το λογισμικό είναι διαθέσιμο στο σύνδεσμο: <https://github.com/madgik/madis>.
- Να αποσυμπίεσετε με unzip το yelp_python.tar.gz.
- Να αλλάξετε το αρχείο settings.py και να βάλετε το τοπικό μονοπάτι που βρίσκεται το madIS στον υπολογιστή σας.
- Να τρέξετε το website.py με την python όπου θα χρησιμοποιήσετε τη γραμμή εντολών.
- Να ανοίξετε κάποιον browser και να βάλετε τη διεύθυνση <http://localhost:9090>.

Γενικά ζητήματα που πρέπει να προσέξετε:

- Ρυθμίστε τον κειμενογράφο που χρησιμοποιείτε έτσι ώστε το tab να ισοδυναμεί με 4 κενά, αλλιώς θα έχετε προβλήματα με τη στοίχιση της Python.
- Θα υπάρχει βαθμολογική μείωση σε περίπτωση που η βάση σας δεν είναι προστατευμένη από sqlinjection.
- Για να μπορεί να διορθωθεί και να βαθμολογηθεί ο κώδικάς σας θα πρέπει τουλάχιστον να μην έχει συντακτικά λάθη. Για αυτό πριν την υποβολή καλό είναι να κάνετε έναν τελευταίο έλεγχο!

Η άσκηση αυτή είναι ομαδική 2-3 ατόμων!

Καλή σας επιτυχία!