Ανάπτυξη λογισμικού για Αλγοριθμικά Προβλήματα

Μάρκος Βαρβαγιάννης

sdi1400017@di.uoa.gr

3η Εργασία

Υλοποίηση αλγορίθμων υπόδειξης κρυπτονομισμάτων (Recommendation) σε C/C++

Η άσκηση αφορά την υλοποίηση αλγορίθμων υπόδειξης (recommendation) με πραγματικά δεδομένα, για την ακρίβεια tweets που αναφέρουν κρυπτονομίσματα. Έχουν υλοποιηθεί σε C++ όλα τα ζητούμενα της εργασίας όπως περιγράφεται στην εκφώνηση, εκτός από το validation.

Μεταγλώτιση

Για να μεταγλωτιστεί το πρόγραμμα τρέξτε **make**

Εκτέλεση

- $./recommendation –d <input file> -o <output file>

Μορφή αρχείων:

Το πρόγραμμα τρέχει στη μορφή των αρχείων input και query που δόθηκαν στο eclass

Αρχεία Κώδικα/Επικεφαλίδες:

- **recommendation.cpp**
  Τα αρχείο που περιέχει τη main
- **recommendation_helping.cpp**, **recommendation_helping.h**
  Βοηθητικές συναρτήσεις για το recommendation όπως οι συναρτήσεις που διαβάζουν το configuration file, το αρχείο με τα vectors, κλπ.
- **dVector.cpp**, **dVector.h**
  Περιέχει τον ορισμό της κλάσης του σημείου - vector
- **hFunction.cpp**, **hFunction.h**
  Περιέχει τον ορισμό της κλάσης συνάρτησης h
- **Cluster.cpp**, **Cluster.h**
  Περιέχει τον ορισμό της κλάσης του Cluster
- **Tweet.cpp**, **Tweet.h**
  Περιέχει τον ορισμό της κλάσης του Tweet
- **lsh.cpp**, **lsh.h**
  Περιέχει τις απαιτούμενες συναρτήσεις για το LSH hashing
- **clustering.cpp**, **clustering.h**
  Συναρτήσεις σχετικά με το clustering
- **parameter_values.h**
  Περιέχει ως defines κάποιες σταθερές, όπως το w και το k για τους αλγόριθμους κοντινών γειτόνων

Input files

Σύμφωνα με την εκφώνηση, δίνεται μόνο ως input το αρχείο με τα tweets. Επομένως τα υπόλοιπα 3 αρχεία που χρειαζόντουσαν (το αρχείο με τα διανυσματικά δεδομένα των tweets μετά το NLP, αυτό με τα cryptocurrencies και αυτό με το λεξικό) ορίζονται ως defines στο **parameter_values.h**. Επομένως, αν θέλετε να αλλάξετε το path των αρχείων, κάντε απλά τις αντίστοιχες αλλαγές στο parameter_values.h.

Δομές που χρησιμοποιήθηκαν

Για τα tweets χρησιμοποιήθηκε ένα **map** το οποίο αντιστοιχίζει τα ids των users με τα tweets τους, αλλά και ένα **unordered_map** στο οποίο αντιστοιχίζονται τα ids των tweets με τα αντίστοιχα αντικείμενα. Τα ίδια τα tweets αρχικοποιούνται μία φορά δυναμικά στη μνήμη για να μην καταλαμβάνουν επιπλέον χώρο.
Για το λεξικό χρησιμοποιήθηκε επίσης ένα map που αντιστοιχεί την κάθε λέξη με το σκορ της.
Για τα κρυπτονομίσματα χρησιμοποιήθηκε ένα απλό vector με strings.
Οι υπόλοιπες δομές έχουν υλοποιηθεί όπως ακριβώς στις προηγούμενες εργασίες χωρίς επιπλέον διορθώσεις

Χρονική διαφορά LSH και clustering

Παρατηρούμε μία σαφή χρονική διαφορά ανάμεσα στον LSH και στο clustering. Το πρώτο θέλει λίγο χρόνο να αρχικοποιηθεί και αρκετή ώρα όταν γίνονται τα queries, ενώ το δεύτερο αντίθετα χρειάζεται χρόνο

στην αρχικοποιήση και στην κατασκευή των clusters, αλλά στα queries είναι γρήγορο. Αυτό είναι λογικό, καθώς το μόνο που χρειάζεται είναι να ανατρέξει στα αντίστοιχα clusters ώστε να κάνει το recommendation και όχι να βρει κοντινότερους γείτονες σε ολόκληρο τον hashtable.

Σημείωση:

- Έχει χρησιμοποιηθεί το εργαλείο git για versioning.