Ανάπτυξη Λογισμικού για Αλγοριθμικά Προβλήματα

Παραδοτέο 1: LSH, Hypercube, KMeans clustering

Μπαρτσώκας Θεόδωρος 1115201700096

Γοργογιάννης Ορέστης 1115201700024

Το παραδοτέο υλοποιήθηκε πλήρως με βάση τις προδιαγραφές της εκφώνησης. Τα αρχεία στα οποία διαμερίστηκε είναι τα εξής:

Util.cpp: Το αρχείο αυτό περιλαμβάνει συναρτήσεις γενικής χρήσεως που χρησιμοποιούνται και από τα 3 προγράμματα που υλοποιήθηκαν. Τέτοιες συναρτήσεις είναι η μετρική manhattan, modular power, συνάρτηση για τον υπολογισμό των s και άλλες.

hash.cpp: Στο αρχείο αυτό βρίσκονται οι κλάσεις του hashtable για lsh και για hypercube. Επίσης, όλες οι μέθοδοι για τις εν λόγω κλασσεις.

lsh_func.cpp: Αυτό το αρχείο περιέχει τις συναρτήσεις που χρησιμοποιεί ο αλγόριθμος lsh και η main που τον υλοποιεί.

hypercube_func.cpp: Αυτό το αρχείο περιέχει τις συναρτήσεις που χρησιμοποιεί ο αλγόριθμος προβολής σε hypercube και η main που τον υλοποιεί.

kmeans_func.cpp: Αυτό το αρχείο περιέχει τις συναρτήσεις που χρησιμοποιεί ο αλγόριθμος kmeans και η main που τον υλοποιεί.

lsh_main.cpp: Το main πρόγραμμα που χρησιμοποιεί lsh για να υλοποιήσει τον αλγόριθμο K-Nearest neighbor και range search.

hypercube_main.cpp: Το main πρόγραμμα που χρησιμοποιεί hypercube για να υλοποιήσει τον αλγόριθμο K-Nearest neighbor και range search.

kmeans_main.cpp: Το main πρόγραμμα που χρησιμοποιεί kmeans clustering με απλό Lloyd's και reverse assignment με range search lsh και hypercube.

Τέλος, τα κατάλληλα αρχεία επικεφαλίδας υλοποιήθηκαν για τις δηλώσεις κλάσεων, struct, μεθόδων και συναρτήσεων.

ΠΕΡΙΓΡΑΦΗ ΛΕΙΤΟΥΡΓΙΑΣ

LSH: Ο αλγόριθμος lsh υλοποιήθηκε κυρίως με χρήση stl vectors. Το κάθε hashtable έχει ένα vector από buckets το οποίο αρχικοποιείται με αριθμό που δίνεται στον constructor. Το bucket είναι υλοποιημένο ως λίστα από struct image, δομή που περιλαμβάνει όλες τις αναγκαίες πληροφορίες για μια εικόνα, όπως ο αριθμός g που της αντιστοιχεί, αλλά και οι αρχικές τιμές των pixel της. Στην κλάση hashtable επίσης αποθηκεύεται ένα vector με τις τιμές των s που αντιστοιχούν στο συγκεκριμένο instance της. Αυτά υπολογίζονται πριν κληθεί ο constructor της κλάσης και του δίνονται σαν όρισμα.

Hypercube: Η υλοποίηση του Hypercube είναι παρόμοια με αυτή του lsh. Η κλάση του έχει δημιουργηθεί μέσω κληρονομικότητας, αφού χρειάζεται σχεδόν όλες τις πληροφορίες του lsh. Η μόνη διαφορά είναι στη συνάρτηση hash, η οποία υπολογίζει τα f(h) και τα κάνει bitwise concatenate για να φτιάξει το g. Η συνάρτηση f(), επιστρέφει τυχαία μια τιμή 0 ή 1 για το κάθε h και στη συνέχεια θυμάται την επιλογή της για μελλοντική χρήση, αποθηκεύοντάς την σε ένα unordered map.

KNN: Ο αλγόριθμος K Nearest Neighbor δέχεται ένα query και το hasháρει στους lsh hashtables ή τον υπερκύβο αντίστοιχα. Στη συνέχεια ελέγχει το bucket που του αντιστοιχεί και αποθηκεύει τις αποστάσεις που βρίσκει σε ένα vector. Στην περίπτωση του υπερκύβου, έχει υπολογίσει τα hamming distances και επισκέπτεται τα buckets με σειρά αύξουσας hamming distance. Τερματίζει όταν κάποιο από τα thresholds των αλγορίθμων φταστεί και επιστρέφει το vector με τις λύσεις, αφού κάνει sort και διαγράψει τα duplicates λόγω πολλαπλών hashtables από το lsh. Τέλος, η main εκτυπώνει στο αρχείο τις πρώτες k.

Range Search: Δουλεύει παρόμοια με τον KNN, με μόνη διαφορά ότι επιστρέφει μόνο τα σημεία με απόσταση κάτω από το δωθέν range και εκτυπώνει όλο το vector που επιστράφηκε.

Kmeans: Το πρόγραμμα αρχικά χρησιμοποιεί τον αλγόριθμο KMeans++ για να αρχικοποιήσει τα K centroids. Στη συνέχεια η ανάθεση γίνεται ανάλογα με τη μέθοδο που έχει επιλέξει ο χρήστης. Αν δεν επιλεγεί μέθοδος, καλείται η απλή Lloyd's. Οι τρεις μέθοδοι διαφέρουν μόνο στην ανάθεση των σημείων σε clusters. Τα range search έγιναν με χρήση του αλγόριθμου από το LSH και το Hypercube. Οι απαραίτητες δομές αρχικοποιούνται κατά την κλήση του αλγόριθμου και διαγράφονται στον τερματισμό του. Η συνθήκη διακοπής που επιλέξαμε είναι σε μία επανάληψη να γίνουν αλλαγές ίσες με μια υποδιαίρεση του αρχικού μεγέθους του αρχείου.

Σχόλια:

Τόσο το w του lsh/hypercube, όσο και το όριο για τον τερματισμό του Kmeans, επιλέχτηκε πειραματικά για να βοηθάει την ακρίβεια χωρίς να θυσιάζει μεγάλο μέρος του χρόνου εκτέλεσης.

Ο χρόνος λειτουργίας είναι στις χειρότερες περιπτώσεις μία τάξη μεγέθους μικρότερος από τον χρόνο του brute force. Ένας μέσος όρος είναι το 2 με 3 τάξεις μεγέθους ανάλογα και το μέγεθος του dataset.