Work Progress

kNN Search with Parallel Incremental Query Answering

Jaouhara Chanchaf

Tuesday Jan 26th, 2023

1. Summary

Done:

AI 1	Find out why Kashif performance degrades when we increase k_max.

AI 0	Measure Kashif recall based on Kashif, PEXESO, JOSIE and LSH Ensemble as our ground truth.
AI 2	Read Progressive search and early termination papers.
AI 3	Search for alternative data structures to store and process knns.

2. Kashif Total NNs insert + sort time

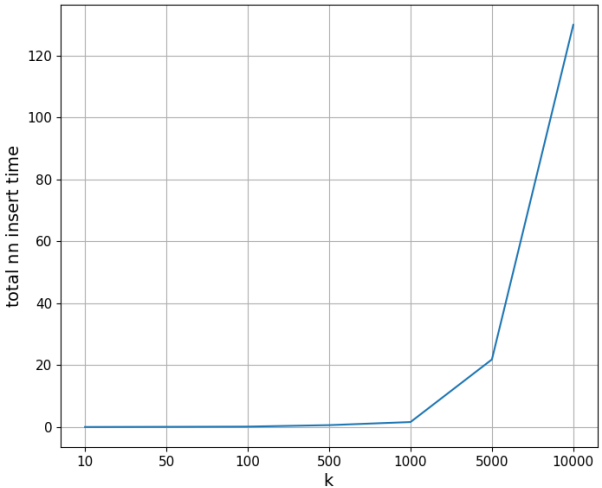

Figure 1: Kashif mean query time

3. Discussion

- Does Kashif require NNs to be sorted, or does it only need the farthest neighbor from the query?
- The alternative data structure must support efficient insert and delete operations and must return the sorted NN list efficiently.
- Max heap supports efficient insert and delete operation. However Heap sort has a complexity of $O(n * log(n))$ and should be performed every time we want to check for a new increment.

4. Approximate Nearest Neighbor search with Adaptive Early Termination (Conlong Li et al., SIGMOD 2020

- **What:** Given a query vector, predict the amount of "work" required to reach the ground truth 1st NN.
- **Why:** State-of-the-art ANN search methods use fixed query independent termination condition(s), which leads to inefficient latency-accuracy tradeoff.
- **How:** Use the query vector and intermediate results as features to train a regression ML model to predict the termination condition.