Graph-Based Vector Search: An Experimental Evaluation of the State-of-the-Art

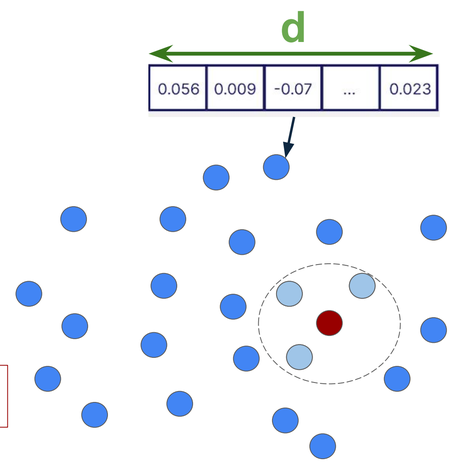
Ilias Azizi[1,2], Karima Echihabi[2], Themis Palpanas[3], Vassilis Christophides[1]

1. ETIS, ENSEA, CNRS, France 2. College of Computing, UM6P, Morocco 3. LIPADE, University of Paris Cité, France

Problem

Vector Search

- **Given:** a set ***S*** of **n** distinct points in **d**-dimensional space R^d with respect to some norm $\| . \|$
- **Goal:** return the set of *K* points in ***S*** that are **closest** to a query $Q \in R^d$, under $\| . \|$

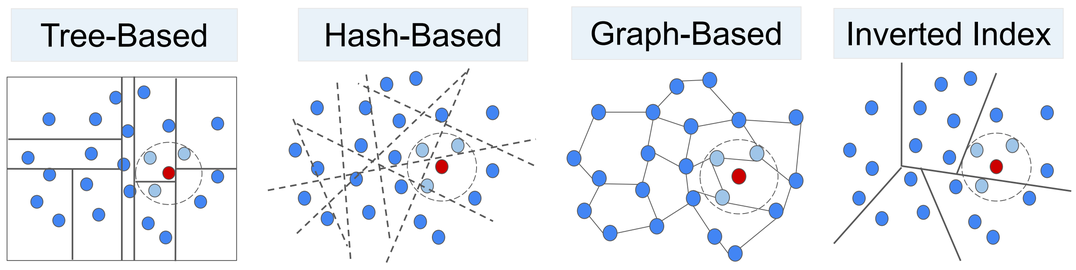


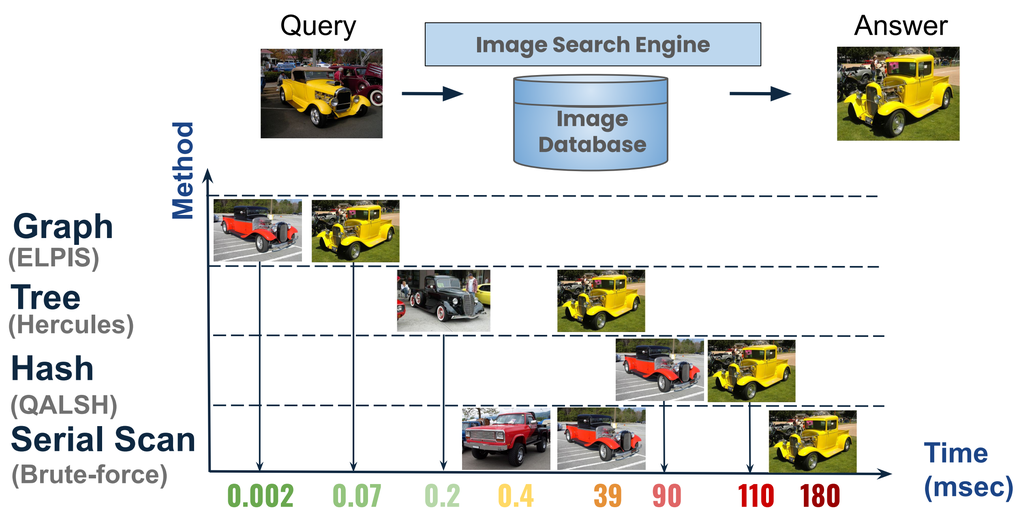
$O(n \cdot d)$

Applications

Forecasting · Classification · Pattern Recognition · Semantic Retrieval · RAG · Object Detection · RecSys · Q&A · Segmentation · Generation

Indexing Approaches





Graph-based Vector Search

- Neighborhood Propagation based
- Incremental Insertion based
- Neighborhood Diversification based
- Divide and Conquer based

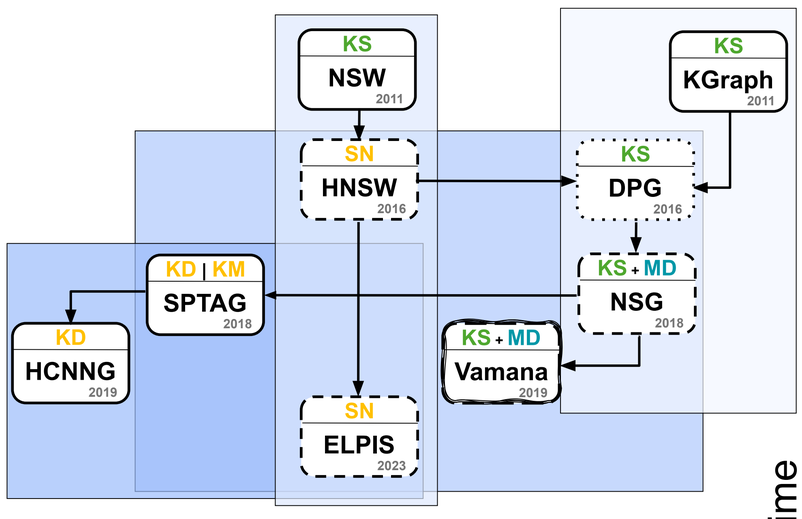
Neighborhood Diversification

NoND · MOND · RND · RRND

Randomized Seed Selection
- K Random Sampling (KS)

Index-based Seed Selection
- Stacked NSW (SN)
- KD, KMean Balanced Trees

Predefined Seed Selection
- Medoid (MD)



Experimental Evaluation

Neighborhood Diversification

Smaller index size and **memory footprint**

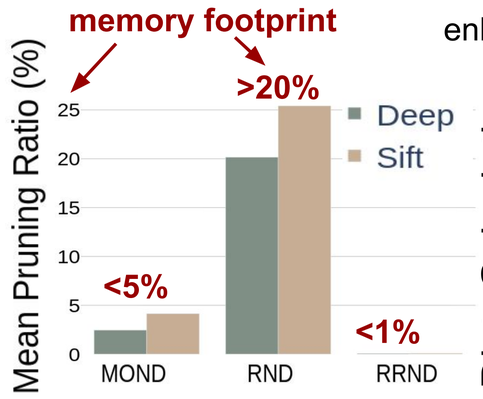


Figure 1. Average edge pruning ratio across nodes

Seed Selection

Optimizing **data structures** for **Seed Selection** enhances **search efficiency** on **large scale datasets**

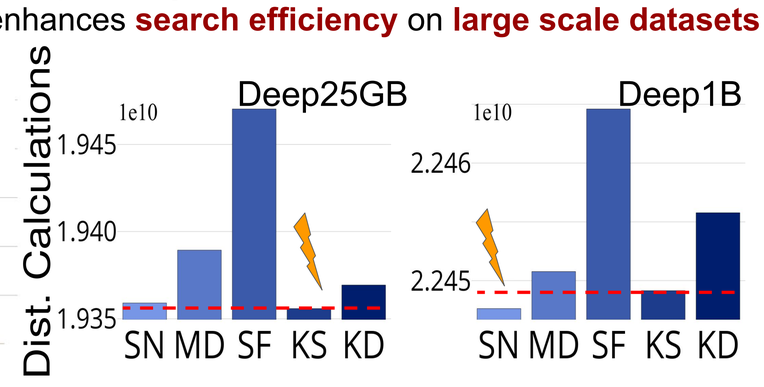


Figure 2. Impact of Seed Selection choice on Search Performance

SOTA Methods Evaluation

ELPIS · HNSW · NSG · VAMANA · DPG · KGRAPH · HCNNG · EFANNA · SPTAG-BKT · SPTAG-KDT · NGT · LSHAPG · SSG

Incremental Insertion based graphs are the most **scalable**

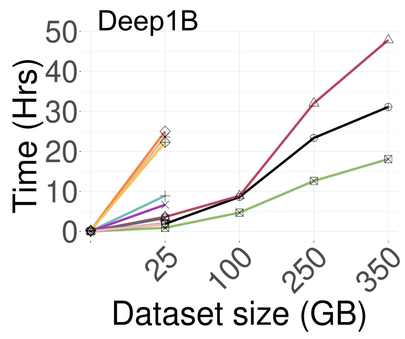


Figure 3. Indexing time

Neighborhood Diversification based graphs are the most **efficient on easy workloads**

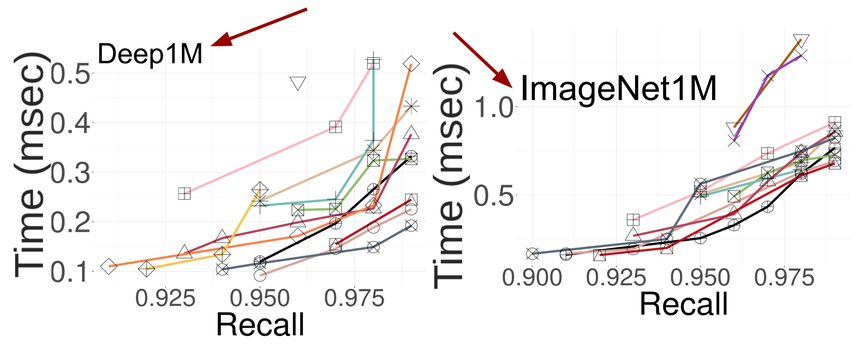


Figure 4. Search Performance on small and easy datasets

Most approaches **fail to scale** efficiently on **large datasets**

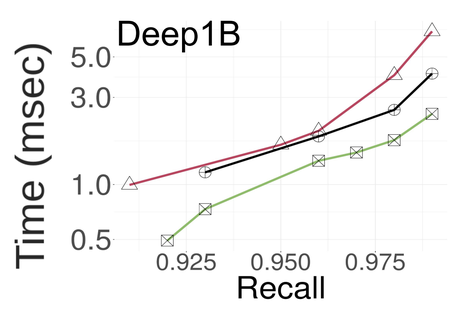


Figure 5. Search Performance on billion scale dataset

Divide-and-Conquer approaches are the most **efficient on hard workloads**

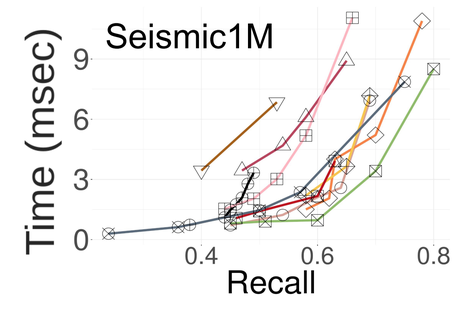
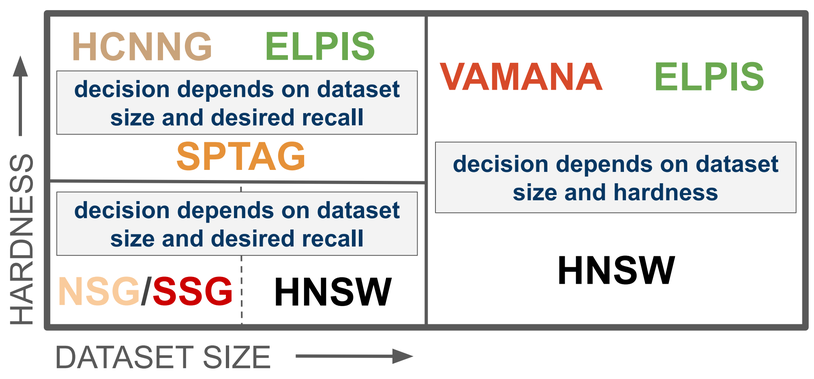


Figure 6. Search Performance on hard dataset

Recommendations

1. Neighborhood Diversification graphs achieve the overall best performance on **small and easy datasets**

2. Divide-and-Conquer and **Relaxed RND** paradigms boost the performance on **hard datasets**



3. Incremental Insertion graph construction offers superior scalability

Équipes Traitement de l'Information et Systèmes · cnrs · UM6P · College of Computing · LIPADE Laboratoire d'Informatique PAris DEscartes · Université Paris Cité