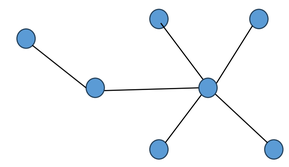
Approximate Single-Linkage Clustering Using Graph-based Indexes: MST-based Approaches and Incremental Searchers

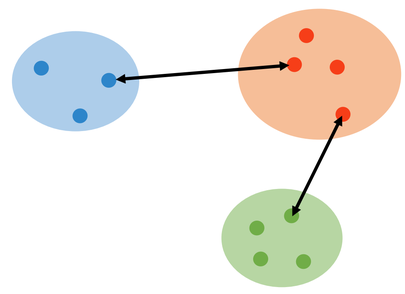
Camilla Birch Okkels, Erik Thordsen††, Martin Aumüller*, Arthur Zimek† and Erich Schubert††*

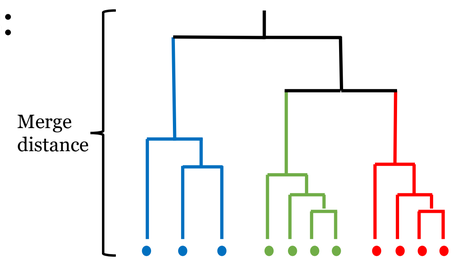
** IT University of Copenhagen, † University of Southern Denmark, †† TU Dortmund*



Single-Linkage

- Single-Linkage:
  - ➢ Continually merge shortest distances.
  - ➢ Equivalent to taking MST of complete graph.
  - ➢ Output a hierarchy of merges:





Naive Kruskal: heap or sort all edges, merge if not connected

- ➢ $O(E)$ memory, $O(E \log(E))$ time, on a complete graph $E = O(n^2)$

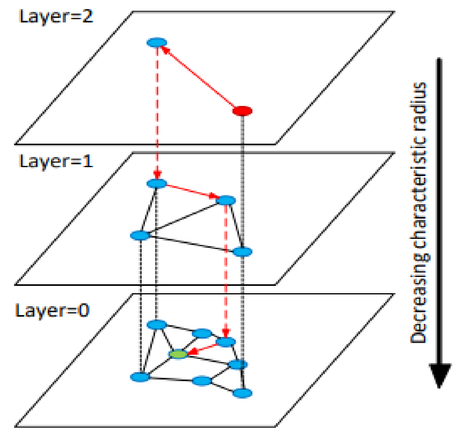
Our question: *How fast and accurate can ANN-based approaches make an approximation of Single-Linkage clustering?*

Techniques

- Incremental search:
  - ➢ Iterator that produces the next nearest neighbor.
  - ➢ Allows stopping or pausing the search when no more neighbors are needed – ultimately avoiding unnecessary distance computations.
- Heap-of-searchers:
  - ➢ Heaps storing current best candidates.
  - ➢ Global heap containing the best neighbor of each node at any given time.
- Hierarchical Navigable Small worlds:
  - ➢ Hierarchy of proximity graphs – higher levels get fewer nodes.
  - ➢ Parameters:

M: max number of edges a node can have.

ef: maximum size of candidate queue.



Schubert, E.: Hierarchical clustering without pairwise distances by incremental similarity search. In: Proc. SISAP. pp. 238–252 (2024)

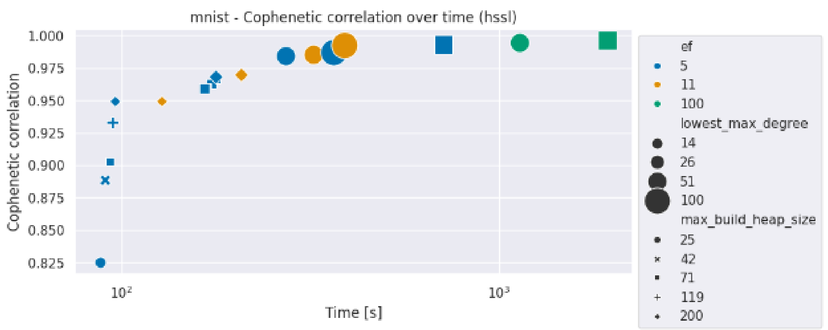
Y. Malkov, D. Yashunin, Efficient and robust approximate nearest neighbor search using Hierarchical Navigable Small World graphs (2016), IEEE Transactions on Pattern Analysis and Machine Intelligence

Algorithms

1. ***HNSWmst***: Compute an MST on the HNSW graph.
2. ***HNSWkruskal***: Compute MST on the HNSW graph, but as edges are merged, add edges of the neighbors of the neighbor not already present to the priority queue.
3. ***HNSWhssl***: Using incremental searchers, traverse the HNSW graph, merge smallest distance from a heap-of-searchers and update searchers as points are merged.

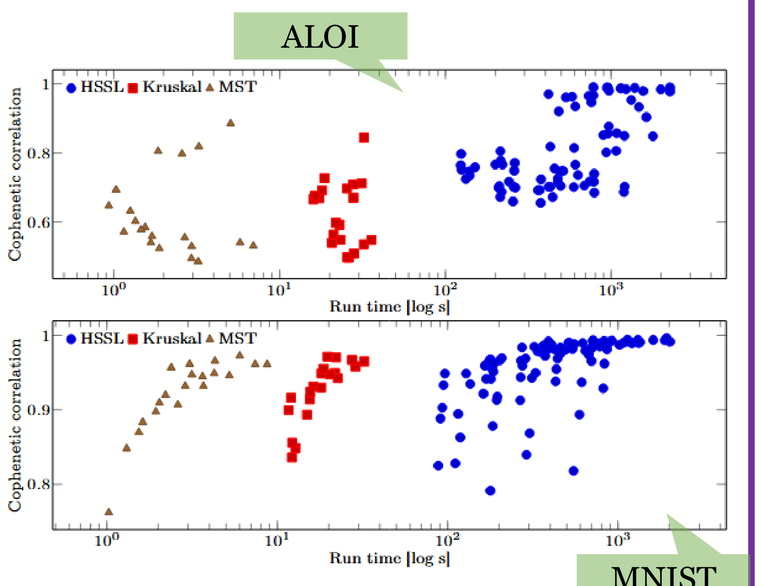
Experimental results

Hyper-parameter optimization



Takeaways:

- Good quality achievable with ef > 5
- All sensitive to hyperparameters, especially MST (On ALOI, more work for less quality in some settings).
- Smallest M tested still gave good quality.



Clustering quality of at least .9 can be achieved in around 2~seconds (MST), 12~seconds (Kruskal), and 100~seconds (HSSL)

Quality and running time

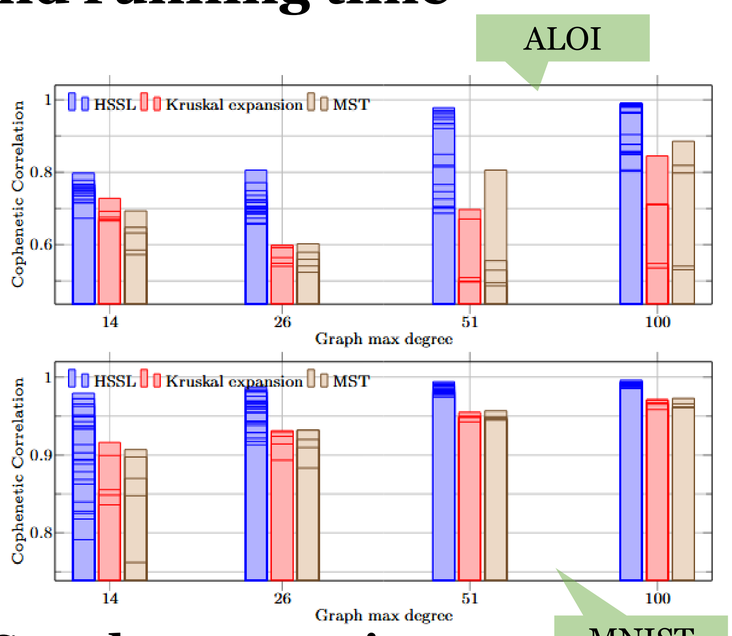
- Great variance in quality.
- Difference in quality small between Kruskal vs. MST.

In general:

HSSL > Kruskal > MST

⬆ M → ⬆ Quality

Fastest settings with quality > 0.8, all faster than SciPy.



Speedup comparison:

MST → (~x10) → Kruskal → (~x5) → HSSL→ (often x2 or more) → SciPy