


CMU SCS

15-826: Multimedia Databases and Data Mining

Lecture #8: Spatial Access Methods - V
Metric trees
C. Faloutsos



CMU SCS

Must-read material

- Textbook, Chapter 5
- Roberto F. Santos Filho, Agma Traina, Caetano Traina Jr., and Christos Faloutsos: [*Similarity search without tears: the OMNI family of all-purpose access methods*](#) ICDE, Heidelberg, Germany, April 2-6 2001. (code at www.cs.cmu.edu/~christos/SRC/OmniUsrKit.tar.gz)

15-826 Copyright: C. Faloutsos (2011) #2




CMU SCS

Outline

Goal: 'Find **similar** / **interesting** things'

- Intro to DB
- ➡ • Indexing - similarity search
- Data Mining

15-826 Copyright: C. Faloutsos (2011) #3




CMU SCS

Indexing - Detailed outline

- primary key indexing
- secondary key / multi-key indexing
- spatial access methods
 - problem dfin
 - z-ordering
 - R-trees
 - misc
- fractals
- text

15-826 Copyright: C. Faloutsos (2011) #4




CMU SCS

SAMs - Detailed outline

- spatial access methods
 - problem dfin
 - z-ordering
 - R-trees
 - misc topics
 - grid files
 - dimensionality curse; dim. reduction
 - metric trees
 - other nn methods
-
- fractals
- text, ...

15-826 Copyright: C. Faloutsos (2011) #5



CMU SCS

Metric trees

- What if we only have a distance function $d(o_1, o_2)$?
- (Applications?)

15-826 Copyright: C. Faloutsos (2011) #6

CMU SCS

Metric trees


- (assumption: $d()$ is a metric: positive; symmetric; triangle inequality)
- then, we can use some variation of ‘Vantage Point’ trees [Yannilos]
- many variations (GNAT trees [Brin95], MVP-trees [Ozsoyoglu+] ...)

15-826 Copyright: C. Faloutsos (2011) #7

CMU SCS

Metric trees

- Finally: M-trees [Ciaccia, Patella, Zezula, vldb 97]
- M-trees = ‘ball-trees’: groups in spheres

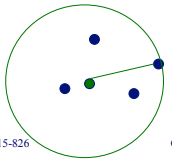


15-826 Copyright: C. Faloutsos (2011) #8


CMU SCS

Metric trees

- Finally: M-trees [Ciaccia, Patella, Zezula, vldb 97]
- M-trees = ‘ball-trees’: Minimum Bounding spheres



15-826 Copyright: C. Faloutsos (2011) #9




CMU SCS

Metric trees

- Search (range and k-nn): like R-trees
- Split?

15-826 Copyright: C. Faloutsos (2011) #10




CMU SCS

Metric trees

- Search (range and k-nn): like R-trees
- Split? Several criteria:
 - minimize max radius (or sum radii)
 - (even: random!)
- Algorithm?

15-826 Copyright: C. Faloutsos (2011) #11



CMU SCS

Metric trees

- Search (range and k-nn): like R-trees
- Split? Several criteria:
 - minimize max radius (or sum radii)
 - (even: random!)
- Algorithm?
- eg., similar to the quadratic split of Guttman

15-826 Copyright: C. Faloutsos (2011) #12

CMU SCS

Metric trees - variations

- Slim trees [Traina+, EDBT2000]
- OMNI tree [Filho+, ICDE2001]

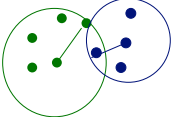
15-826 Copyright: C. Faloutsos (2011) #13

CMU SCS

Metric trees - Slim trees

- How to improve the structure?

BEFORE



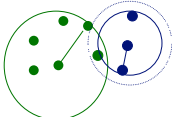
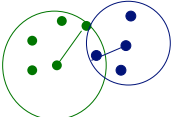
15-826 Copyright: C. Faloutsos (2011) #14

CMU SCS

Metric trees - Slim trees

- Idea: give-away contents, if it decreases the radius - eg:

BEFORE AFTER




15-826 Copyright: C. Faloutsos (2011) #15

CMU SCS

Metric trees - Slim trees

- How to accelerate the splitting time
- ($O(N^2)$, currently)?




15-826 Copyright: C. Faloutsos (2011) #16

CMU SCS

Metric trees - Slim trees

- Split using Minimum Spanning Tree (drop longest edge)




15-826 Copyright: C. Faloutsos (2011) #17

CMU SCS

Metric trees - Slim trees

- Split using Minimum Spanning Tree (drop longest edge)




15-826 Copyright: C. Faloutsos (2011) #18

CMU SCS

Metric trees - Slim trees

- Split using Minimum Spanning Tree (drop longest edge)



15-826 Copyright: C. Faloutsos (2011) #19

CMU SCS

Metric trees - Slim trees

- result: at least as fast as M-trees for search
- MST: significantly faster for split, with tiny performance penalty

15-826 Copyright: C. Faloutsos (2011) #20

CMU SCS

SAMs - Detailed outline

- spatial access methods
 - problem defn
 - z-ordering
 - R-trees
 - misc topics
 - grid files
 - dimensionality curse; dim. reduction
 - metric trees
- fractals
- text, ...

15-826 Copyright: C. Faloutsos (2011) #21

CMU SCS

Metric trees - OMNI trees


- How to turn objects into vectors?
- (assume that distance computations are expensive; we need to answer range/nn queries quickly)

15-826 Copyright: C. Faloutsos (2011) #22

CMU SCS

Metric trees - OMNI trees

- How to turn objects into vectors?
- A: pick n 'anchor' objects; record the distance of each object from them $\rightarrow n$ -d vector




15-826 Copyright: C. Faloutsos (2011) #23

CMU SCS

Metric trees - OMNI trees

- How to turn objects into vectors?
- A: pick n 'anchor' objects; record the distance of each object from them $\rightarrow n$ -d vector

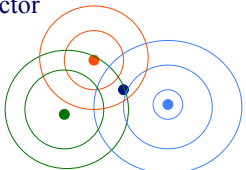


15-826 Copyright: C. Faloutsos (2011) #24

CMU SCS

Metric trees - OMNI trees

- How to turn objects into vectors?
- A: pick n 'anchor' objects; record the distance of each object from them $\rightarrow n$ -d vector

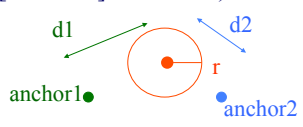


15-826 Copyright: C. Faloutsos (2011) #25

CMU SCS

Metric trees - OMNI trees

- we could put OMNI coordinates in R-tree (or other SAM, or even do seq. scan)
- and still answer range and nn queries! (see [Filho'01] for details)

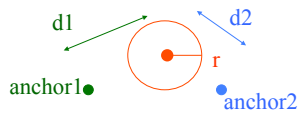


15-826 Copyright: C. Faloutsos (2011) #26


CMU SCS

Metric trees - OMNI trees

- Result: faster than M-trees and seq. scanning (especially if distance computations are expensive)



15-826 Copyright: C. Faloutsos (2011) #27




CMU SCS

Metric trees - OMNI trees

- Q1: how to choose anchors?
- Q2: ... and how many?

15-826 Copyright: C. Faloutsos (2011) #28




CMU SCS

Conclusions for SAMs

- z-ordering and R-trees for low-d points and regions
- M-trees & variants for metric datasets
- beware of the ‘dimensionality curse’

15-826 Copyright: C. Faloutsos (2011) #29




CMU SCS

References

- Aurenhammer, F. (Sept. 1991). “Voronoi Diagrams - A Survey of a Fundamental Geometric Data Structure.” ACM Computing Surveys 23(3): 345-405.
- Bentley, J. L., B. W. Weide, et al. (Dec. 1980). “Optimal Expected-Time Algorithms for Closest Point Problems.” ACM Trans. on Mathematical Software (TOMS) 6(4): 563-580.
- Burkhard, W. A. and R. M. Keller (Apr. 1973). “Some Approaches to Best-Match File Searching.” Comm. of the ACM (CACM) 16(4): 230-236.

15-826 Copyright: C. Faloutsos (2011) #30




CMU SCS

References

- Christian Böhm, Stefan Berchtold, Daniel A. Keim: [*Searching in high-dimensional spaces: Index structures for improving the performance of multimedia databases*](#). ACM Comput. Surv. 33(3): 322-373 (2001)
- Edgar Chávez, Gonzalo Navarro, Ricardo A. Baeza-Yates, José L. Marroquín: [*Searching in metric spaces*](#). ACM Comput. Surv. 33(3): 273-321 (2001)

15-826 Copyright: C. Faloutsos (2011) #31




CMU SCS

References

- Ciaccia, P., M. Patella, et al. (1997). M-tree: An Efficient Access Method for Similarity Search in Metric Spaces. VLDB.
- Filho, R. F. S., A. Traina, et al. (2001). Similarity search without tears: the OMNI family of all-purpose access methods. ICDE, Heidelberg, Germany.
- Friedman, J. H., F. Baskett, et al. (Oct. 1975). "An Algorithm for Finding Nearest Neighbors." IEEE Trans. on Computers (TOC) C-24: 1000-1006.

15-826 Copyright: C. Faloutsos (2011) #32




CMU SCS

References

- Fukunaga, K. and P. M. Narendra (July 1975). "A Branch and Bound Algorithm for Computing k-Nearest Neighbors." IEEE Trans. on Computers (TOC) C-24(7): 750-753.
- Shapiro, M. (May 1977). "The Choice of Reference Points in Best-Match File Searching." Comm. of the ACM (CACM) 20(5): 339-343.

15-826 Copyright: C. Faloutsos (2011) #33

CMU SCS

References

- Shasha, D. and T.-L. Wang (Apr. 1990). "New Techniques for Best-Match Retrieval." ACM TOIS 8(2): 140-158.
- Traina, C., A. J. M. Traina, et al. (2000). Slim-Trees: High Performance Metric Trees Minimizing Overlap Between Nodes. EDBT, Konstanz, Germany.

15-826

Copyright: C. Faloutsos (2011)

#34
