Are You My Neighbor? Bringing Order to Neighbor Computing Problems.

David C. Anastasiu[1,2] , Huzefa Rangwala[3], and Andrea Tagarelli[4]

[1]Computer Engineering, San Jose State University, CA

[1]Computer Science & Engineering, Santa Clara University, CA

[2]Computer Science & Engineering, George Mason University, VA

[3]DIMES, University of Calabria, Italy

25th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, August 2019

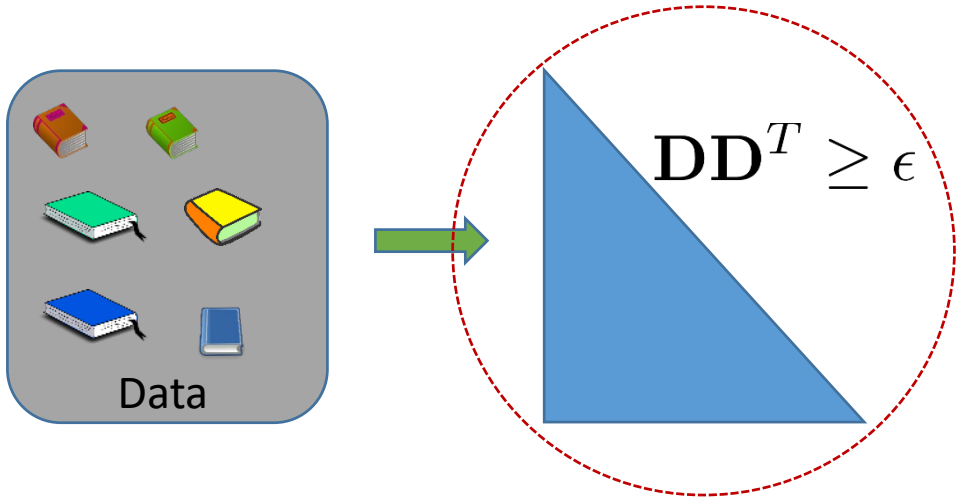
Tutorial Outline

- **Part I: Problems and Data Types**
  - Dense, sparse, and asymmetric data
  - Bounded nearest neighbor search
  - Nearest neighbor graph construction
  - Classical approaches and limitations
- **Part II: Neighbors in Genomics, Proteomics, and Bioinformatics**
  - Mass spectrometry search
  - Microbiome analysis
- **Part III: Approximate Search**
  - Locality sensitive hashing variants
  - Permutation and graph-based search
  - Maximum inner product search
- **Part IV: Neighbors in Advertising and Recommender Systems**
  - Collaborative filtering at scale
  - Learning models based on the neighborhood structure
- **Part V: Filtering-Based Search**
  - Massive search space pruning by partial indexing
  - Effective proximity bounds and when they are most useful
- **Part VI: Neighbors in Learning and Mining Problems in Graph Data**
  - Neighborhood as cluster in a complex network system
  - Neighborhood as influence trigger set

Part I:
Problems and Data Types

David C. Anastasiu, San José State University [david.anastasiu@sjsu.edu]

Starting September:
Department of Computer Science and Engineering
Santa Clara University



Similarities

Data Mining / Machine Learning

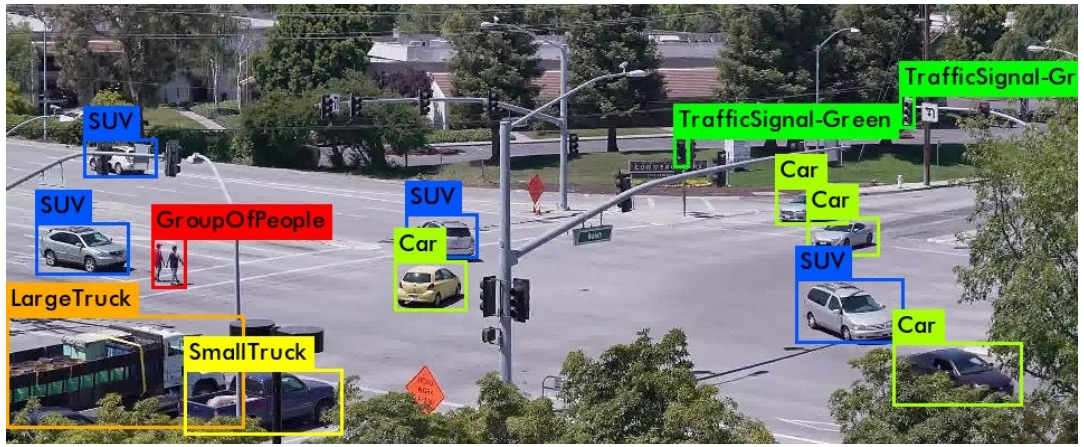
- Threat/Outlier detection
- Duplicate detection
- Advertising/Recommendation
- Clustering
- Pharmaceutical virtual screening
- etc…

Traffic Analytics

[IEEE CVPR'19] CityFlow: A City-Scale Benchmark for Multi-Target Multi-Camera Vehicle Tracking and Re-Identification
[IEEE SOSE'19, IEEE SOSE'19, IEEE MC'19, IEEE CVPRW'18, IEEE SmartWorld'17]

w/ Milind Naphade, CTO of AI Cities, NVIDIA

- Organizing member and Evaluation Chair for the AI City Challenge.
- Address challenges in traffic analysis from video, including:
  - Multi-camera vehicle tracking
  - Speed estimation from video
  - Anomaly detection



Document and Text Mining

[Flex Grant] Industry collaboration.

Detect fraud and/or savings opportunities in expense reports

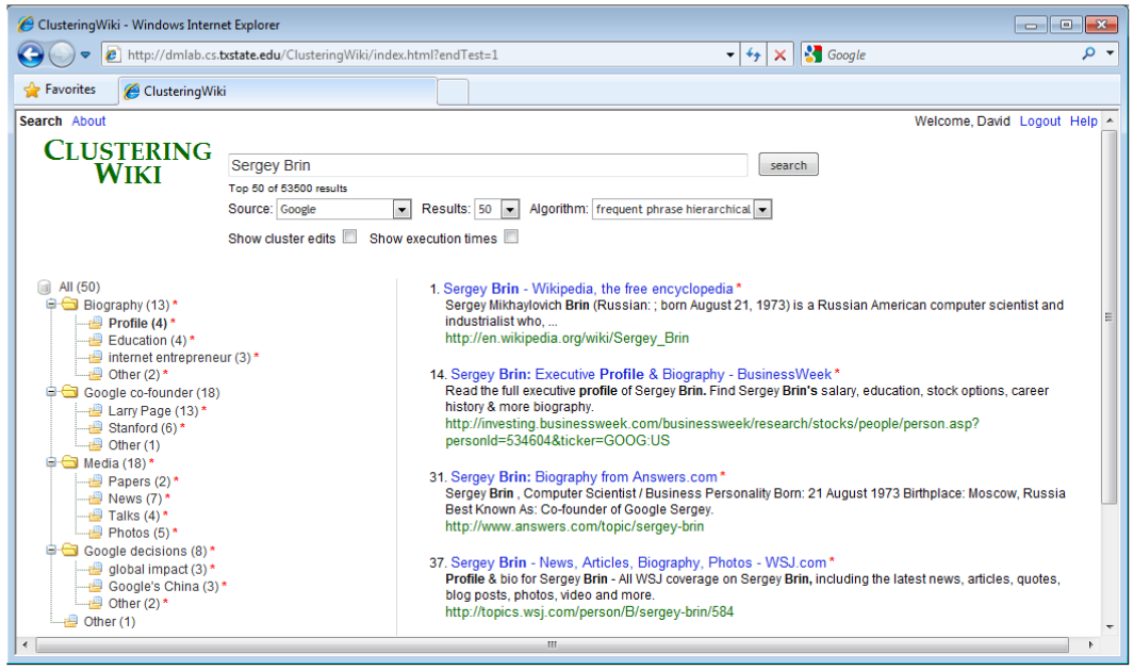
- Receipt localization and classification (ResNet, YOLO-like models)
- Object character recognition (CNN + Bi-directional LSTM)
- Knowledge extraction (NER, heuristics)
- Report-receipt matching (KNN using visual features)

Hard problem, complicated by the multinational and multilingual aspect of the Flex business

Improve Quality and Utility of Search Results

[CIKM'11] A Framework for Personalized and Collaborative Clustering of Search Results

- Developed a "label-first" hierarchical clustering technique.
- Path-based collaborative editing of cluster labels and assignments.
- Method incorporated in a document processing pipeline at LLNL.



Lawrence Livermore National Laboratory

Open Modification Spectral Library Search

[Grant NSF 1850557] CRII: III: RUI: Effective Protein Characterization via Fast Exact Open Modification Searching

w/ William Stafford Noble, Genome Sciences, UW

- Methods for characterizing the protein composition of biological samples
  - Mass spectrometers output relative abundance histograms (spectra)
  - Massive databases exist for protein-associated spectra (spectral libraries)
  - Task is to match unknown spectra against nearest neighbor in library

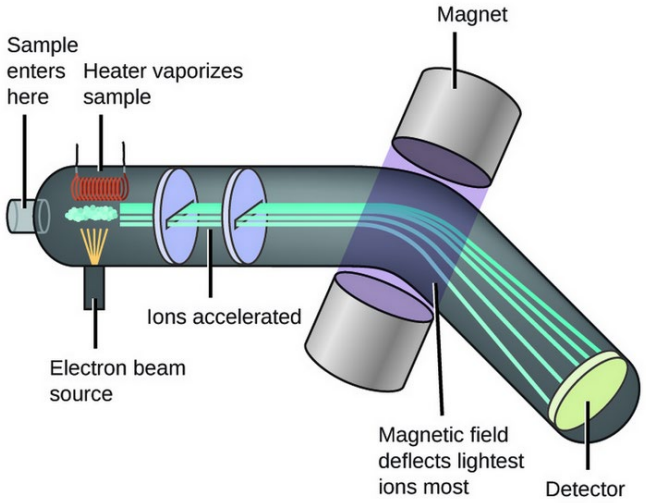
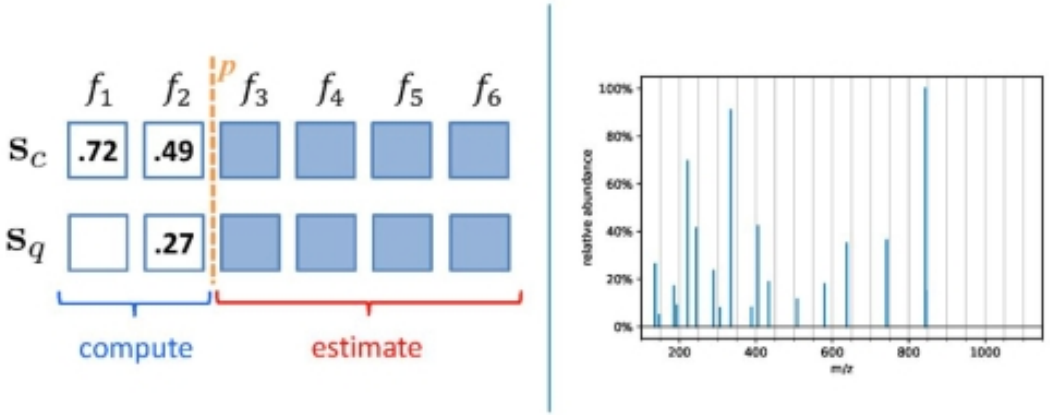


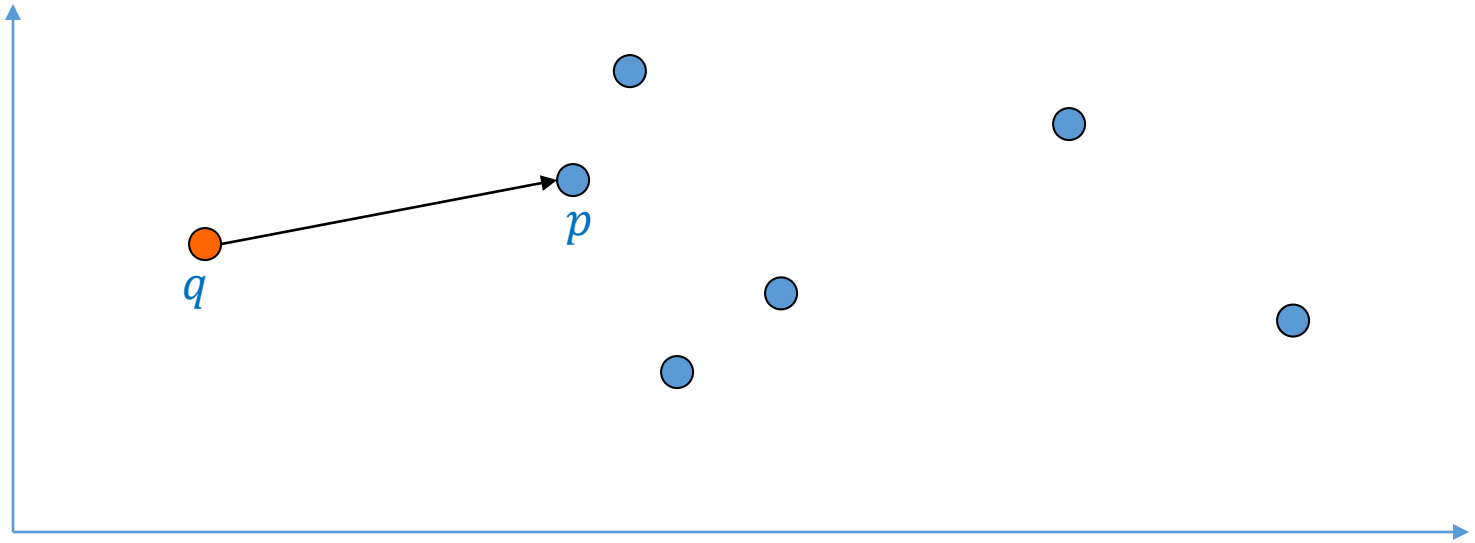
Image: https://i.stack.imgur.com/iVYVY.png

- Challenges
  - Imperfect ionization/spectrometry
  - Size of databases (10's to 100's or million)



Nearest Neighbor Search

- *Given:* a set P of n points in $\mathbb{R}^d$
  a query point q from a set Q in $\mathbb{R}^d$
- *Goal:* find the *nearest neighbor* p of q in P



What are we searching for?

- Object representation assigns meaning to object features
  - Continuous / discrete / binary
  - Synchronous / asynchronous
  - Dense / sparse
  - High- / low-dimensional

- Most of the time we represent objects as point vectors in some Euclidean space
  - Lends itself to easily-understood algebraic and geometric relationships between the points

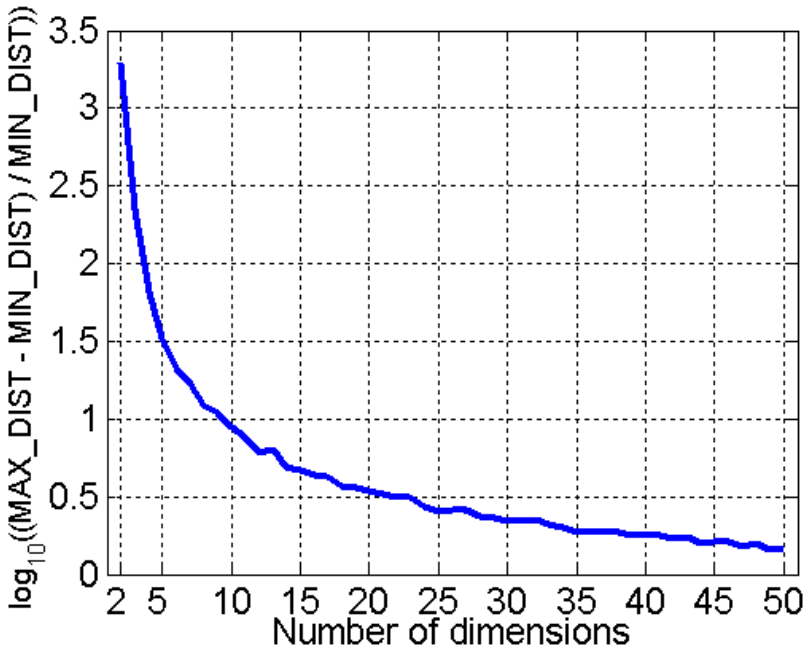
Asymmetric attributes

- Only presence (a non-zero attribute value) is regarded as important
  - Words present in documents
  - Items present in customer transactions
  - Edges in a graph
- If we met a friend in the grocery store would we ever say the following?
  "I see our purchases are very similar since we didn't buy most of the same things."
- Asymmetric attributes typically arise from objects that are sets.
- They lend themselves to a *sparse* vector representation
  - More than 50% of values are 0's, and 0's are ignored

Curse of dimensionality

- When dimensionality increases, data becomes increasingly sparse in the space that it occupies.

- Definitions of density and distance between points, which is critical for clustering and outlier detection, become less meaningful.
  - Still meaningful if there is structure in the data, i.e., points are clustered/grouped



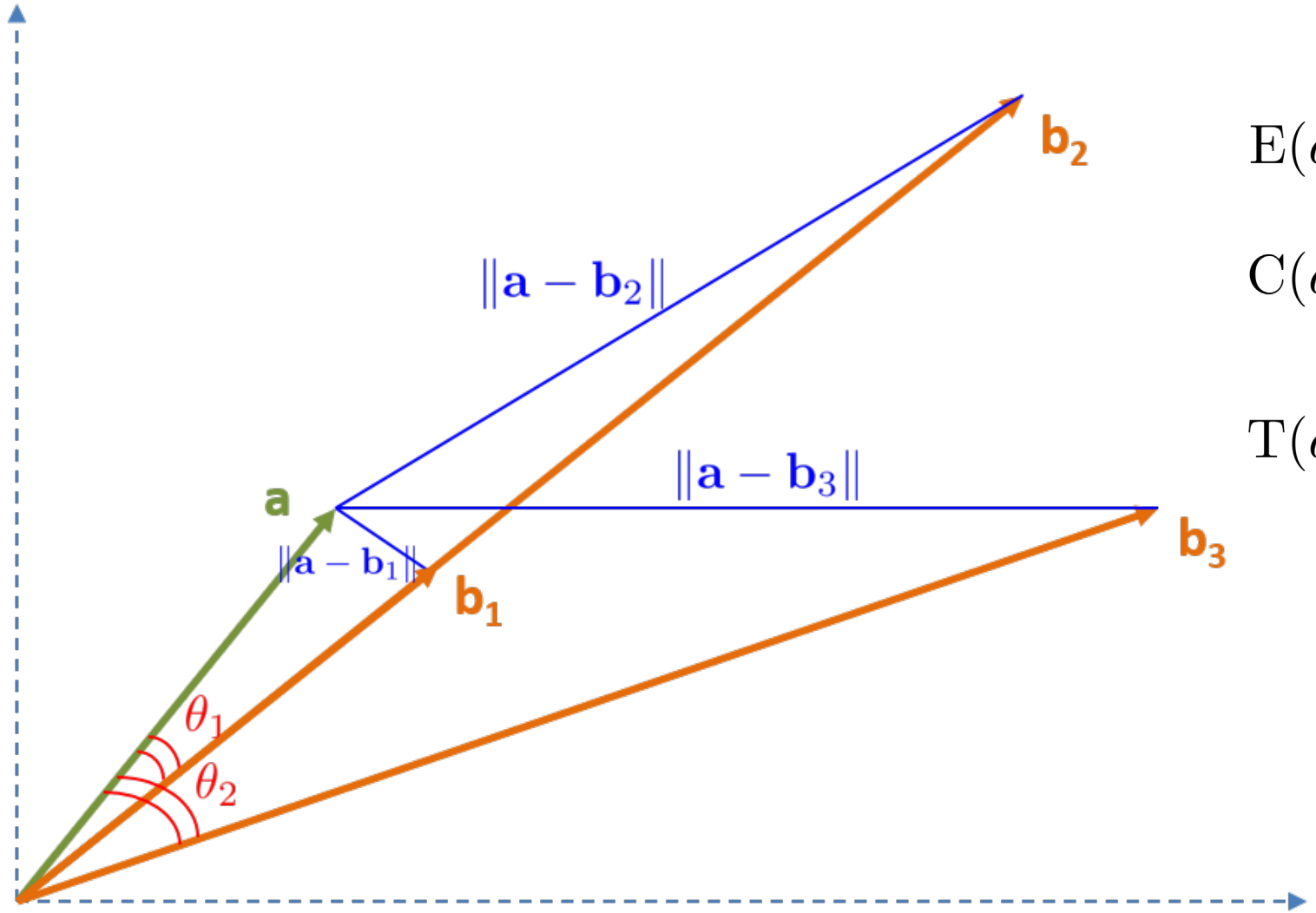
- Randomly generate 500 points.
- Compute difference between max and min distance between any pair of points.

Dimensionality Reduction

- Purpose:
  - Avoid curse of dimensionality.
  - Reduce amount of time and memory required by data mining algorithms.
  - Allow data to be more easily visualized.
  - Help to eliminate irrelevant features or reduce noise.

- Techniques:
  - Principal Components Analysis (PCA)
  - Singular Value Decomposition
  - Others: supervised and non-linear techniques

How do we decide the objects are close?



$$\mathrm{E}(d_i, d_j) = ||\mathbf{d}_i - \mathbf{d}_j||_2$$

$$\mathrm{C}(d_i, d_j) = \frac{\langle \boldsymbol{d}_i, \boldsymbol{d}_j \rangle}{||\mathbf{d}_i||_2 \times ||\mathbf{d}_j||_2}$$

$$\mathrm{T}(d_i, d_j) = \frac{\langle \boldsymbol{d}_i, \boldsymbol{d}_j \rangle}{||\mathbf{d}_i||_2^2 + ||\mathbf{d}_j||_2^2 - \langle \boldsymbol{d}_i, \boldsymbol{d}_j \rangle}$$

Bounded nearest-neighbor search

Extensions of the nearest neighbor problem:

- k-nearest neighbor (NN) search
  - Find k closest neighbors
- Min-ϵ similarity search (max-r distance, or radius search)
  - Find all neighbors with similarity $\geq \epsilon$ (within distance r from the query)
- k-nearest neighbor graph construction
  - Find k closest neighbors for all objects in the set
- All-pairs similarity search (min-ϵ graph construction)
  - Find all neighbors with similarity $\geq \epsilon$ for all the objects in the set

Background on NN methods

- Tree-based methods (small dimensionality)
  - Quad-tree, K-d-tree, VP-tree, R-tree (and variants), Cover-tree, PCA-tree, Ball-tree, K-means tree, Spill-tree, HD-index, etc.
- Stochastic methods (approximate)
  - LSH (and variants), C2LSH, QALSH, FLANN, KGraph, ANNOY, LEMP, FAISS, FLASH, EFANNA, KIFF, HNSW, NGT, etc.
- Filtering based methods (exact)
  - All-Pairs, MMJoin, APT, L2AP, CANN, L2Knng, TAPNN, etc.

- Tutorial focused on NNs for real-valued vectors
  - Esp. for Euclidean distance and cosine similarity
  - Esp. high-dimensional vectors

References

Tree-based methods:

[Quad-tree] Finkel, R. A.; Bentley, J. L. (1974). "Quad Trees A Data Structure for Retrieval on Composite Keys". Acta Informatica. Springer-Verlag. 4: 1–9.

[K-d-tree] J. K. Friedman, J. L. Bentley, and R. A. Finkel, “An algorithm for finding best matches in logarithmic expected time,” ACM Tran. on Mathematical Software, Vol. 3, No. 3, 209–226 (1977).

[VP-tree] Nielsen, Frank (2009). "Bregman vantage point trees for efficient nearest Neighbor Queries". Proceedings of Multimedia and Exp (ICME). IEEE. pp. 878–881.

[R-tree] A. Guttman, “R-trees: A dynamic index structure for spatial searching,” in: Proc. ACM SIGMOD ICMD’84 (1984), pp. 47–57.

[Cover-tree] Kenneth Clarkson. Nearest-neighbor searching and metric space dimensions. In G. Shakhnarovich, T. Darrell, and P. Indyk, editors, Nearest-Neighbor Methods for Learning and Vision: Theory and Practice, pages 15--59. MIT Press, 2006.

[PCA-tree] R. F. Sproull, “Refinements to nearest-neighbor searching in k-dimensional trees,” Algorithmica, Vol. 6, No. 1, 579–589 (1991).

[Ball-tree] S. M. Omohundro, Five Balltree Construction Algorithms, ICSI TR-89-063 (1989).

[K-means tree] K. Fukunaga and P. M. Narendra, “A branch and bound algorithm for computing k-nearest neighbors,” IEEE Trans. Comput., Vol. C-24, No. 7, 750–753 (1975).

[Spill-tree] T. Liu, A. W. Moore, A. Gray, and K. Yang, “An investigation of practical approximate nearest neighbor algorithms,” in: Proc. NIPS’04 (2004), pp. 825–832.

[HD-index] Akhil Arora, Sakshi Sinha, Piyush Kumar, and Arnab Bhattacharya. 2018. HD-index: pushing the scalability-accuracy boundary for approximate kNN search in high-dimensional spaces. Proc. VLDB Endow. 11, 8 (April 2018), 906-919. DOI: https://doi.org/10.14778/3204028.3204034

References

Stochastic/Approximate methods:

Locality-Sensitive Hashing:

[LSH] P. Indyk and R. Motwani. Approximate nearest neighbors: towards removing the curse of dimensionality. In STOC, pages 604–613, 1998.

[MinHash] Broder, Andrei Z.; Charikar, Moses; Frieze, Alan M.; Mitzenmacher, Michael (1998), "Min-wise independent permutations", Proc. 30th ACM Symposium on Theory of Computing (STOC '98), New York, NY, USA: Association for Computing Machinery, pp. 327–336

[BayesLSH] Venu Satuluri and Srinivasan Parthasarathy. 2012. Bayesian locality sensitive hashing for fast similarity search. Proc. VLDB Endow. 5, 5 (January 2012), 430-441.

[E2LSH] M. Datar, N. Immorlica, P. Indyk, and V. S. Mirrokni. Locality-sensitive hashing scheme based on p-stable distributions. In SCG, pages 253–262, 2004.

[C2LSH] J. Gan, J. Feng, Q. Fang, and W. Ng. Locality-sensitive hashing scheme based on dynamic collision counting. In SIGMOD, pages 541–552, 2012.

[FLANN] Marius Muja and David G. Lowe: "Scalable Nearest Neighbor Algorithms for High Dimensional Data". Pattern Analysis and Machine Intelligence (PAMI), Vol. 36, 2014

[QALSH] Q. Huang, J. Feng, Y. Zhang, Q. Fang, and W. Ng. Query-aware locality-sensitive hashing for approximate nearest neighbor search. PVLDB, 9(1):1–12, 2015.

Quantization:

[PQ] H. Jegou, M. Douze, and C. Schmid. Product quantization for nearest neighbor search. PAMI, 33(1):117–128, 2011.

[OPQ] T. Ge, K. He, Q. Ke, and J. Sun. Optimized product quantization for approximate nearest neighbor search. In CVPR, pages 2946–2953, 2013.

References

Graph traversal:

[KGraph] Wei Dong, Charikar Moses, and Kai Li. Efficient k-nearest neighbor graph construction for generic similarity measures. In Proceedings of the 20th International Conference on World Wide Web, WWW '11, pages 577–586, New York, NY, USA, 2011. ACM.

[HNSW] Y. A. Malkov and D. A. Yashunin. Efficient and robust approximate nearest neighbor search using hierarchical navigable small world graphs. arXiv:1603.09320 [cs.DS], 2016.

[EFANNA] Cong Fu and Deng Cai. EFANNA: An Extremely Fast Approximate Nearest Neighbor Search Algorithm Based on kNN Graph, arXiv: 1609.07228(2016).

[NSG] Cong Fu and Chao Xiang and Changxu Wang and Deng Cai. "Fast Approximate Nearest Neighbor Search With The Navigating Spreading-out Graphs" PVLDB 12(5):461-474, 2019.

[KIFF] A. Boutet, A. Kermarrec, N. Mittal and F. Taiani, "Being prepared in a sparse world: The case of KNN graph construction," 2016 IEEE 32nd International Conference on Data Engineering (ICDE), Helsinki, 2016, pp. 241-252.

Mixed/Other:

[NGT] Iwasaki, M., Miyazaki, D.: Optimization of Indexing Based on k-Nearest Neighbor Graph for Proximity. arXiv:1810.07355 [cs] (2018).

[ANNOY] https://github.com/spotify/annoy

[LEMP] Christina Teflioudi, Rainer Gemulla, and Olga Mykytiuk. 2015. LEMP: Fast Retrieval of Large Entries in a Matrix Product. In Proceedings of the 2015 ACM SIGMOD International Conference on Management of Data (SIGMOD '15). ACM, New York, NY, USA, 107-122. DOI: https://doi.org/10.1145/2723372.2747647

GPU:

[FLASH] Y. Wang, A. Shrivastava, and J. Ryu, FLASH: Randomized Algorithms Accelerated over CPU-GPU for Ultra-High Dimensional Similarity Search. arXiv:1709.01190. 4 Sep 2017.

[FAISS] Johnson, Jeff and Douze, Matthijs and Jegou, Herve. "Billion-scale similarity search with GPUs" arXiv preprint arXiv:1702.08734, 2017

References

Filtering-based methods

[All-Pairs] Roberto J. Bayardo, Yiming Ma, and Ramakrishnan Srikant. 2007. Scaling up all pairs similarity search. In Proceedings of the 16th international conference on World Wide Web (WWW '07). ACM, New York, NY, USA, 131-140.

[MMJoin] Dongjoo Lee, Jaehui Park, Junho Shim, and Sang-goo Lee. 2010. An efficient similarity join algorithm with cosine similarity predicate. In Proceedings of the 21st international conference on Database and expert systems applications: Part II (DEXA'10), Pablo Garcia Bringas, Abdelkader Hameurlain, and Gerald Quirchmayr (Eds.). Springer-Verlag, Berlin, Heidelberg, 422-436.

[APT] A. Awekar and N. F. Samatova, “Fast matching for all pairs similarity search,” in Proceedings of the 2009 IEEE/WIC/ACM International Joint Conference on Web Intelligence and Intelligent Agent Technology - Volume 01, ser. WI-IAT ’09. Washington, DC, USA: IEEE Computer Society, 2009, pp. 295–300.

[L2AP] David C. Anastasiu and George Karypis. L2AP: Fast Cosine Similarity Search With Prefix L-2 Norm Bounds. Proceedings of the 30th IEEE International Conference on Data Engineering (ICDE 2014).

[TAPNN] David C. Anastasiu and George Karypis. Efficient Identification of Tanimoto Nearest Neighbors. Proceedings of the 3rd IEEE International Conference on Data Science and Advanced Analytics (DSAA 2016).

[CANN] David C. Anastasiu. Cosine Approximate Nearest Neighbors. In Data Science -- Analytics and Applications (iDSC 2017), pages 45-50, Springer Fachmedien Wiesbaden, 2017.

[L2Knng] David C. Anastasiu and George Karypis. L2Knng: Fast Exact K-Nearest Neighbor Graph Construction with L2-Norm Pruning. In 24th ACM International Conference on Information and Knowledge Management, CIKM '15, 2015.

[TA-like] Yuliang Li, Jianguo Wang, Benjamin Pullman, Nuno Bandeira and Yannis Papakonstantinou. Index-based, High-dimensional, Cosine Threshold Querying with Optimality Guarantees. ICDT 2019.