MSc Internship – Stage M2 2017-18: Big data on a budget: Investigating compact representations for non-parametric KNN graph construction

1 Supervision

Advisers	François Taiani, `francois.tainai@irisa.fr`, 02 99 84 75 04
	Olivier Ruas, `olivier.ruas@inria.fr`
Lab	IRISA (UMR 6074)
Team	ASAP (As Scalable As Possible)
	Équipes-Projet Inria / Département D1 IRISA

2 Context

K-Nearest-Neighbor (KNN) graphs[1] play a fundamental role in many big data applications, including numerous on-line services such as search [1, 2], recommendation [3, 8, 9] and classification [10]. A KNN graph is a directed graph of entities (e.g., users, products, services, documents etc.), in which each entity (or *node*) is connected to its k most similar counterparts or *neighbors*, according to a given *similarity metric*. In a large number of applications, this similarity metric is computed from a second set of entities (termed *items*) associated with each node in a bipartite graph (often extended with weights, such as ratings or frequencies). For instance, in a movie rating database, nodes are users, and each user is associated with the movies (items) she has already rated [5].

A KNN graph can be obtained using a *brute force* approach by computing the similarity between every pair of nodes. The resulting KNN graph is exact, but the process rapidly becomes intractable on large datasets: under a brute force strategy, a dataset with millions of nodes—a common instance—requires trillions of similarity computations.

Many applications, however, value speed over accuracy, and only require a good approximation of the KNN graph [6, 7]. Recent KNN construction algorithms [3, 4] have therefore sought to reduce the number of similarity computations by exploiting an *incremental greedy strategy*. These approaches starts from an initial random graph that is iteratively refined towards an approximate nearest neighbors (ANN) graph. These techniques repeatedly check each node's neighbors' neighbors to improve the quality of their own neighborhood, thus dramatically reducing the number of comparisons between nodes.

Greedy algorithms are among the most efficient approximate KNN approaches to date. They seem, however, to have reached their limits, and it is now difficult to see how to further reduce the number of comparisons pursuing this paradigm.

3 Objective

The ASAP team is currently working on an orthogonal approach to the acceleration of KNN construction based on *fingerprints*, i.e. a *fixed-size* and *binary* representation of the items associated with an entity. Our scheme uses a simple uniform 1-bit hashing technique, related to Bloom filters. Our experiments show this basic approach is able to deliver up to a 82% reduction in computation speed, while incurring only a small loss in KNN quality.

This loss is caused by collisions inherent to the hashing technique. **The goal of this internship is to investigate how these collisions could be minimized to increase the quality of the resulting KNN for a given fingerprint length.** A higher quality would also make it possible to further reduce fingerprints, and improve performance.

[1] Note that the problem of computing a complete KNN graph (which we address in this paper) is related but different from that of answering a sequence of KNN queries, a point we revisit when discussing related work.

4 Tasks

- Perform a bibliographic study of existing binary compact representation schemes that can be used for similarity estimation, and their computation cost.
- Explore the impact of collisions of Single Hash Fingerprints (SHF) on several representative (entity ×items) datasets.
- Investigate the use of non-uniform hashing techniques to reduce the impact of collisions on similarity estimation, KNN construction, and an application use case (recommendation).
- If time permits, investigate the use of foldable non-uniform bit-arrays and stochastic binarization to provide varying levels of compression depending on the size of and details of profiles.

References

[1] Xiao Bai, Marin Bertier, Rachid Guerraoui, Anne-Marie Kermarrec, and Vincent Leroy. Gossiping personalized queries. In *EDBT*, pages 87–98, 2010.

[2] Marin Bertier, Davide Frey, Rachid Guerraoui, Anne-Marie Kermarrec, and Vincent Leroy. The gossple anonymous social network. In *Proceedings of the ACM/IFIP/USENIX 11th International Conference on Middleware*, pages 191–211. Springer-Verlag, 2010.

[3] Antoine Boutet, Davide Frey, Rachid Guerraoui, Anne-Marie Kermarrec, and Rhicheek Patra. Hyrec: leveraging browsers for scalable recommenders. In *Proceedings of the 15th International Middleware Conference*, pages 85–96. ACM, 2014.

[4] 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*, pages 577–586. ACM, 2011.

[5] F. Maxwell Harper and Joseph A. Konstan. The movielens datasets: History and context. *ACM Trans. Interact. Intell. Syst.*, 5(4):19:1–19:19, December 2015.

[6] Kimberly Keeton, Charles B Morrey III, Craig AN Soules, and Alistair Veitch. Lazybase: freshness vs. performance in information management. *ACM SIGOPS Operating Systems Review*, 44(1):15–19, 2010.

[7] Alexandros Labrinidis and Nick Roussopoulos. Exploring the tradeoff between performance and data freshness in database-driven web servers. *The VLDB JournalâThe International Journal on Very Large Data Bases*, 13(3):240–255, 2004.

[8] J. J. Levandoski, M. Sarwat, A. Eldawy, and M. F. Mokbel. Lars: A location-aware recommender system. In *ICDE*, 2012.

[9] Greg Linden, Brent Smith, and Jeremy York. Amazon. com recommendations: Item-to-item collaborative filtering. *Internet Computing, IEEE*, 7(1):76–80, 2003.

[10] N. Nodarakis, S. Sioutas, D. Tsoumakos, G. Tzimas, and E. Pitoura. Rapid aknn query processing for fast classification of multidimensional data in the cloud. *CoRR*, abs/1402.7063, 2014.