文章编号：2095-2163(2022)06-0001-06 中图分类号：TP311.13 文献标志码：A

近邻问题的亚线性算法研究现状综述

马恒钊[1]，李建中[1,2]
（1 哈尔滨工业大学 海量数据计算研究中心，哈尔滨 150001；2 中科院深圳理工大学，广东 深圳 518107）

摘 要：大数据时代已经来临，海量数据计算要求设计亚线性算法。本文选择了大数据分析问题中比较重要的问题、即近邻问题，包括近似最近邻问题、近似 k-最近邻问题以及全 k-最近邻问题，对其亚线性算法的研究现状做了综述。

关键词：全 k-最近邻；近似最近邻；近似 k-最近邻；亚线性算法

A survey on the sub-linear algorithms about the Nearest Neighbors problems

MA Hengzhao[1], LI Jianzhong[1,2]
(1 Research Institute of Massive Data Computing, Harbin Institute of Technology, Harbin 150001, China;
2 Shenzhen University of Science and Technology, Chinese Academy of Sciences, Shenzhen Guangdong 518107, China)

【Abstract】 The era of big data has come, and massive data computing requires the design of sub-linear algorithms. This paper chooses several important problems in big data analysis, namely the Nearest Neighbors problems, including the All-k-Nearest Neighbors, Approximate Nearest Neighbors and Approximate k-Nearest Neighbors problem, and provides a survey on the sub-linear algorithms about these problems.

【Key words】 All-k-Nearest Neighbors; Approximate Nearest Neighbors; Approximate k-Nearest Neighbors; sub-linear algorithms

0 引 言

近年来，大数据概念已经在研究界和应用界越来越热门，这也表明大数据时代已然来临。许多应用已经开始日常处理起 TB 级数据，比如广为人知的 TeraSort 应用[1]。而在一些场景、比如科学数据中，甚至开始面对 PB、以及 EB 级数据，诸如大型综合巡天望远镜（Large Synoptic Survey Telescope，LSST）[2]每天生成的数据量就达到 1.25 PB。下面做一个简单的计算。设想要通过 SSD 固态硬盘来读取数据，目前 SSD 的最大读带宽约为 6 GB/s，于是，仅读取 1 PB 数据就需要 34.7 h，读取 1 EB 数据甚至需要超过 4 年时间。这表明，在处理大数据时，串行线性算法的运行时间也有可能是不可接受的。如何高效地处理这样天文级数量的数据，成为理论研究界和应用界共同的挑战。应用界提出的解决办法一般是并行化，比如淘宝应对双十一的海量事务处理请求所用的就是阿里云等并行处理平台。但是从理论界看来，并行化有以下一些问题[3-4]：

（1）有些问题是无法高效并行化的。

（2）并行化并不能降低问题的复杂度。

（3）并行化带来了通信复杂度和通信瓶颈等问题。

（4）这也是最重要的一项，并行化有可扩展性和加速比的问题，即当所使用的处理器数量达到一定阈值时，再增加处理器将无法降低总运行时间，甚至还有可能会增加总运行时间。因此，理论界解决大数据问题所提出的方案是设计亚线性算法，即时间复杂度为 $O(log^k n)$ 或者 $O(n^\rho)$ 的算法，其中 $k > 0$，$\rho < 1$。只有设计亚线性算法，才能从根本上降低算法的时间复杂度，减小处理大数据所需要的时间。

根据文献[5]提出的理论，亚线性算法分为 2 类。其一为纯亚线性算法，即可以直接通过亚线性算法解决的问题；其二为伪亚线性算法，即经过一个多项式时间的预处理后，再通过亚线性算法可以解决的问题。前者包括判定数组是否为 ε-far 单调问题，具体算法参见文献[6]。后者则包括众所周知的无序数组查找问题，即在无序数组上查找元素，可以通过花费 $O(nlogn)$ 时间进行排序，再通过 $O(1)$ 时间的二分查找予以解决。亚线性算法已经有接近二十年的研究历史，最初的研究内容集中于属性测

基金项目：国家自然科学基金(61832003)。

作者简介：马恒钊(1995-)，男，博士研究生，主要研究方向：亚线性算法、近邻问题；李建中(1950-)，男，教授，博士生导师，主要研究方向：数据库、无线传感网、大数据计算理论与管理等。

收稿日期：2021-03-26

试[6-7],后来又拓展至图算法[8-9]、计算几何[10]、代数计算[9]等领域。参见综述文献[11]。这些问题在许多领域都有广泛的应用,比如生物信息[12]、物联网[13]、轨迹分析[14]、机器学习[15]、推荐系统[16]等。然而对于近似最近邻领域,其亚线性算法的研究却还未臻充分。本文对该问题的亚线性算法的研究现状做了综述。

1 全 k-最近邻问题算法概述

全 k-最近邻问题简记作 All-kNN。关于此问题的工作最早可以追溯到 1983 年[17],且近年来也一直有关于此问题的研究工作出现[18-19]。此问题备受各方关注的原因,是因为许多应用都以 All-kNN 问题作为重要的子程序,比如分类[20]、凝聚式聚类[21]、图像检索[22]、推荐系统[23]以及离群点检测[24]等。在许多类似的应用中,计算 All-kNN 都是主要的瓶颈[21]。

由于该问题的重要性,历史上研究者们提出了许多算法试图高效地解决 All-kNN 问题,这些算法可以分为以下 3 类:

(1)第一类算法。使用各种不同的技巧来降低算法的实际运行时间,然而这些算法的最坏时间复杂度仍为 $O(n^2)$ 不变。据研究分析可知,这些算法大致有 3 种算法设计技巧。分述如下。

① 第一种,基于树型的空间索引。如 $k-d$ 树[25]和 Voronoi 图[26]等。

② 第二种,基于空间填充曲线,包括希尔伯特曲线[27]和 Z-order 曲线[28]等。空间填充曲线是一类在高维数据上建立一维索引的方法,基于空间填充曲线构建的索引具有一种重要特性,即在空间上相近的点更容易被分配到相近的索引项中。据文献[19, 29-30]中的结果,此性质有助于降低计算 All-kNN时需要计算距离的次数。

③ 第三种,基于近邻传播的思想,即近邻的近邻仍很有可能是近邻。NN-descent 算法[31]是提出此思想的开创性工作,且仍是目前 All-kNN 问题最好的算法之一。其他使用此思想的算法一般先以某种方法作为预处理,再使用近邻传播技巧来提高结果的精度。例如,文献[32]中使用随机分治方法作为预处理,而文献[33]中使用局部敏感哈希方法作为预处理。

(2)第二类算法。是在并行环境下解决问题。文献[34]理论上证明了 All-kNN 问题存在并行最优算法,该算法在 CREW PRAM 模型上需要 $O(logn)$ 时间和 $O(n)$ 个处理器。其他工作则致力于在不同的并行平台上解决 All-kNN 问题,比如 MapReduce 平台[35-36]和 GPU 环境[37]。

上述算法的最坏时间复杂度上界都为 $O(n^2)$,第三类算法则不同,现已都被证明了具有更低的时间复杂度,且都是串行算法,与文献[34]中给出的并行算法也不同。例如,Bentley[38]给出了一种多维分治算法,可以在 $O(n\ (1)^{d-1})$ 时间内解决 All-kNN 问题,其中 d 是数据的维数。又如,文献[17]中给出的算法需要 $O(n(log\delta))$ 时间,其中 δ 是输入数据点集中最远的一对点和最近的一对点的距离之比。再者,文献[39]中给出的算法声明具有 $O(kd^dnlogn)$ 的时间复杂度上界。最后,研究发现了文献[39]中算法的一个错误,在文献[40]中给出了严格的证明,证明文献[39]中的算法的下界为 $\Omega(n^2)$,并提出了新的算法,真正具有 $O(nlogn)$ 时间上界。

2 近似最近邻问题算法概述

近似最近邻问题,简记作 ε-NN,是一个在理论研究和应用研究方面都很重要、且基础性的问题,自 19 世纪 90 年代起就有许多相关的解决算法被提出,这些算法可以分为 4 类,如下所述。

(1)第一类算法,试图直接解决问题,且这些算法一般都设计了预处理数据结构来支持算法的高效运行。Arya 等人[12]研究提出了一种算法,需要 $1/\varepsilon^{O(d)}\cdot n$ 空间,$1/\varepsilon^{O(d)}\cdot nlogn$ 预处理时间以及 $1/\varepsilon^{O(d)}\cdot logn$ 查询时间。另有研究[13]提出的算法需要 $O(dn)$ 空间、$O(dnlogn)$ 预处理时间和 $(d/\varepsilon)^{O(d)}\cdot logn$ 查询时间。Kleinberg 在文献[14]中提出了 2 个算法。其一是确定性算法且达到了 $O(d\,log^2d(d+logn))$ 查询时间,使用的数据结构需要 $O((nlogd)^{2d})$ 空间和 $O((nlogd)^{2d})$ 预处理时间;其二是随机化算法,预处理时间为 $O(d^2\,log^2d\cdot n\,log^2n)$,所需的空间降至为 $O(dn\cdot log^3n)$ 但其查询时间却提升为 $O(dn\cdot log^3n)$。

(2)第二类算法,考虑 $\varepsilon=d^{O(1)}$ 的特殊情况。文献[15]中给出了这种情况下的一个算法,可以在 $O(2^dlogn)$ 查询时间内回答 $O(\sqrt{d})$-NN,且需要 $O(d^8dnlogn)$ 预处理时间和 $O(d\,2^dn)$ 空间。基于此,Chan[16]改进了上述结果,提出了一个可以在 $O(d^2logn)$ 查询时间内回答 $O(d^{3/2})$-NN 的算法,将需要 $O(d^2nlogn)$ 预处理时间和 $O(dnlogn)$ 空间。

(3)第三类算法,试图从另一个角度考虑ε-NN问题。算法利用了数据的一种内生维度,而非原始数据所存在的欧氏空间的维度。文献[17]给出了一个代表性的工作,该文献中提出的算法的查询时间复杂度上界为$2^{O(dim(P))}log\Delta+(1/\varepsilon)^{O(dim(P))}$,其中$dim(P)$为输入点集$P$的内生维度,$\Delta$是$P$的直径。

在上述3类算法之外,Indyk等人[18]开创了第四类算法。这类算法的关键思想在于定义了一个近似近邻问题,记作(c,r)-NN,然后将ε-NN问题归约到此问题。于是解决ε-NN问题的过程就分为了2部分,一是解决(c,r)-NN问题,二是设计从ε-NN到(c,r)-NN的归约。目前,有3个现存工作设计了从ε-NN到(c,r)-NN的归约算法。对此拟做阐释分析如下。

(1)第一个归约算法是文献[18]中提出的,所构造的数据结构需要$O(n\cdot polylog(n))$空间,其查询时间为$O(log^2n)$。

(2)第二个归约算法[19]将查询时间降低到$O(logn/\varepsilon)$,其预处理时间和空间都为$O\left(dn\frac{logn}{\varepsilon}log\frac{n}{\varepsilon}\right)$。

(3)第三个算法[20]在预处理阶段调用(c,r)-NN算法来构造数据结构,其查询时间为$O(log^{O(1)}n)$,此处的指数$O(1)$来自于算法的常数成功概率的要求。这3个现存算法的查询时间都是比较高的,目前已在文献[41]中提出了新的图灵归约算法,具体查询时间为$O(1)$,低于所有上述现存算法。

3　近似k-最近邻问题算法概述

近似k-最近邻问题,简记作kANN问题,是近似最近邻问题的自然推广,已在许多应用领域都有重要应用,比如数据可用性[42-44]、数据库查询[45]、以及图算法[46]等。对于kANN问题有2种近似标准,分别称作距离标准和召回率标准。其中,距离标准要求近似结果集中的点到查询点的最远距离和精确结果集中的点到查询点的最远距离之比不超过给定阈值。召回率标准要求近似结果集和精确结果集的交集大小不小于给定阈值。下面将简要讨论现存工作是如何考虑上述2个近似标准的。

关于kANN问题的现存算法可以分为4类。这里给出探讨剖析如下。

(1)第一类算法,是基于树的方法。这种方法的主要思想是将度量空间递归地划分为子空间,并组织成树结构。K-D树[47]是这种思想的代表性工作,但是这种方法只在低维空间中有效,当维度数升高时其性能会大幅下降。Vantage point树[48]是另一种基于树的方法,在方法性能上对K-D树有所提升。FLANN方法[49]是最近的工作且在高维空间中有更好的表现,但是却有可能输出非最优的结果[50]。据研究所知,现存的基于树的方法对距离标准和近似标准都没有理论保证。

(2)第二类算法,是基于置换的方法。方法的思想是挑选一个枢轴点的集合,并将每个数据元素表示成这些枢轴点的一个置换,这个置换是通过将枢轴点按照和数据元素的距离排序来得到的。在这种表示方法中,距离较近的数据元素的置换表示也相似。利用这种思想的方法包括MI-File[51]和PP-Index[52]。然而,据分析可知,这些方法也没能给出对距离标准和召回率标准的理论保证。

(3)第三类算法,是基于局部敏感哈希(Locality Sensitive Hashing,LSH)的方法。LSH最初由Indyk等人[18]提出,并应用于解决$k=1$时的kANN问题。不久之后,Datar等人[53]提出了第一个实用的LSH函数,此后关于LSH的理论和应用研究就大量出现[54-55]。例如,Andoni等人[56]证明了基于LSH的算法的最优时空下界,并提出了符合下界的最优的LSH函数[57]。在应用方面,Gao等人[58]提出了一种致力于弥合LSH的理论和kANN应用的算法。可以参考概述文献[59]。基本的LSH算法只能在$k=1$的情况下满足距离标准[60]。一些较晚近的算法有更多的进展。比如C2LSH[61]解决了距离标准下的kANN问题,但是却要求近似因子必须是整数的平方。SRS算法[62]也是针对距离标准下kANN问题的算法,但是却只有部分的理论保证,即只有算法在特定条件下结束时,返回的结果才能满足距离标准。

(4)第四类算法,是基于图的算法。在这类算法中用到的特殊的图称为近似图,其中的边是基于顶点之间的几何关系定义的。可以参考概述文献[63]。例如,Paredes等人[64]使用了kNN图,Ocsa等人[65]使用的是相对邻居图(Relative Neighborhood Graph,RNG),Malkov等人[66]使用的是可导航的小世界图(Navigable Small World Graph,NSW)。在这种基于图的kANN算法中经常用到在近似图上的导航过程,即选择图上的某一个顶点作为起始点,并按照某种特定的导航策略来向着目标顶点移动。据分析可知这种上述的这些方法都无法在理论上满足距

离标准和召回率标准。

总之,大部分现存算法在距离标准和召回率标准上都没有理论近似保证。在现存工作中,召回率标准只被用于度量实验结果的好坏程度,距离标准则只被少数算法部分地满足[61-62]。故在文献[67]中提出了将 2 个近似标准结合起来的新问题,并提出了针对该问题的算法,通过理论分析,证明了在输入服从泊松点过和的情况下,算法返回的结果能够至少满足其中一个近似标准,并证明了算法的期望预处理时间复杂度为 $O(nlogn)$,期望空间复杂度为 $O(nlogn)$,期望查询时间复杂度为 $O(1)$。

4 结束语

在本文中,研究回顾了近邻问题中的全 k-最近邻问题、近似最近邻问题及近似 k-最近邻问题,主要介绍了这些问题现有算法的时间复杂度。从中可以看出,这些问题的现有算法的时间复杂度都无法达到亚线性,因此在可接受的时间内将无法在 PB 级大数据上求得计算结果。研究者应该更多关注于亚线性时间算法的设计与分析,从而能够应对大数据时代的高效计算需求。

参考文献

[1] Nyberg C, Shah M. Sortbenchmark home page[EB/OL]. [2021]. http://sortbenchmark.org/.

[2] IVEZIĆ, KAHN S M, TYSON J A, et al. LSST: From science drivers to reference design and anticipated data products[J]. The Astrophysical Journal, 2019, 873(2): 111.

[3] GREENLAW R, HOOVER H J, RUZZO W L, et al. Limits to parallel computation: P-completeness theory[M]. New York, USA: Oxford University Press on Demand, 1995.

[4] CENSOR Y, ZENIOS S A. Parallel optimization: Theory, algorithms, and applications[M]. New York, USA: Oxford University Press on Demand, 1997.

[5] GAO X, LI J, MIAO D, et al. Recognizing the tractability in big data computing[J]. Theoretical Computer Science, 2020, 838: 195-207.

[6] ERGÜN F, KANNAN S, KUMAR S R, et al. Spot-checkers[C]// Proceedings of the thirtieth annual ACM symposium on Theory of computing - STOC'98. New York, USA: ACM Press, 1998: 259-268.

[7] PARNAS M, RON D. Testing the diameter of graphs[J]. Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 1999, 1671: 85-96.

[8] FEIGE U. On sums of independent random variables with unbounded variance and estimating the average degree in a graph[J]. SIAM Journal on Computing, 2006, 35(4): 964-984.

[9] CHAZELLE B, RUBINFELD R, TREVISAN L. Approximating the minimum spanning tree weight in sublinear time[J]. SIAM Journal on Computing, 2005, 34(6): 1370-1379.

[10] CHAZELLE B, LIU Ding, MAGEN A. Sublinear geometric algorithms[J]. SIAM Journal on Computing, 2005, 35(3): 627-646.

[11] RUBINFELD R, SHAPIRA A. Sublinear time algorithms[J]. SIAM Journal on Discrete Mathematics, 2011, 25(4): 1562-1588.

[12] ARYA S, MOUNT D M. Approximate nearest neighbor queries in fixed dimensions[C]// Proceedings of the Fourth Annual ACM/SIGACT-SIAM Symposium on Discrete Algorithms. Austin, Texas, USA: SIAM, 1993: 271-280.

[13] KRIEGER S, KECECIOGLU J D. Boosting the accuracy of protein secondary structure prediction through nearest neighbor search and method hybridization[J]. Bioinform., 2020, 36(Supplement-1): i317-i325.

[14] KLEINBERG J M. Two algorithms for nearest-neighbor search in high dimensions[C]//Proceedings of the Twenty-Ninth Annual ACM Symposium on Theory of Computing-STOC'97. New York, USA: ACM Press, 1997: 599-608.

[15] QUEK Y T, WOO W L, THILLAINATHAN L. IoT Load classification and anomaly warning in ELV DC picogrids using hierarchical extended k-nearest neighbors[J]. IEEE Internet Things J., 2020, 7(2): 863-873.

[16] CHAN T M. Approximate nearest neighbor queries revisited[C]// Proceedings of the Thirteenth Annual Symposium on Computational Geometry-SCG'97. New York, USA: ACM Press, 1997, 20: 352-358.

[17] WANG Sheng, BAO Zhifeng, CULPEPPER J S, et al. Reverse k nearest neighbor search over trajectories[J]. 2018 IEEE 34th International Conference on Data Engineering (ICDE), 2018, 30(4): 757-771.

[18] INDYK P, MOTWANI R. Approximate nearest neighbors: Towards Removing the Curse of Dimensionality[C]//Proceedings of the Thirtieth Annual ACM Symposium on Theory of Computing-STOC'98. New York, USA: ACM Press, 1998: 604-613.

[19] LI Hao, LIU Xiaojie, LI Tao, et al. A novel density-based clustering algorithm using nearest neighbor graph[J]. Pattern Recognition, 2020, 102: 107206.

[20] JAYALAKSHMI N, PADMAJA P, SUMA G J. Webpage recommendation system using interesting subgraphs and Laplace based k-nearest neighbor[J]. Int. J. Pattern Recognit. Artif. Intell., 2020, 34(3): 2053003:1-2053003:30.

[21] CLARKSON K L. Fast algorithms for the all nearest neighbors problem[C] // 24th Annual Symposium on Foundations of Computer Science (SFCS 1983): UCSON, AZ, USA: IEEE, 1983, 16: 226-232.

[22] PARK Y, LEE S G. A novel algorithm for scalable k-nearest neighbor graph construction[J]. Journal of Information Science, 2016, 42(2): 274-288.

[23] SIERANOJA S. High dimensional k NN-graph construction using space filling curves[D]. Finland: University of Eastern Finland, 2015.

[24] SZUMMER M, JAAKKOLA T. Partially labeled classification with Markov random walks[C]//NIPS'01: Proceedings of the 14th International Conference on Neural Information Processing Systems: Natural and Synthetic. Cambridge, MA, USA: MIT Press, 2001: 945-952.

[25] FRANTI P, VIRMAJOKI O, HAUTAMAKI V. Fast agglomerative

clustering using a k-nearest neighbor graph[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2006, 28(11): 1875-1881.

[26] YANG Xingwei, LATECKI L J. Affinity learning on a Tensor product graph with applications to shape and image retrieval[C]// CVPR'11: Proceedings of the 2011 IEEE Conference on Computer Vision and Pattern Recognition. Washington, DC, USA: IEEE Computer Society, 2011:2369-2376.

[27] KARYPIS G. Evaluation of item-based top-N recommendation algorithms[C]// Proceedings of the Tenth International Conference on Information and Knowledge Management-CIKM'01. New York, USA: ACM Press, 2001:247-254.

[28] BRITO M, CHÁVEZ E, QUIROZ A, et al. Connectivity of the mutual k-nearest neighbor graph in clustering and outlier detection[J]. Statistics & Probability Letters, 1997, 35(1): 33-42.

[29] FRIEDMAN J H, BENTLEY J L, FINKEL R A. An algorithm for finding best matches in logarithmic expected time[J]. ACM Transactions on Mathematical Software (TOMS), 1977, 3(3): 209-226.

[30] EDELSBRUNNER H. CONSTRUCTING ARRANGEMENTS: Algorithms in combinatorial geometry[M]//Algorithms in Combinatorial Geometry. EATCS Monographs in Theoretical Computer Science. Berlin/Heidelberg: Springer, 1987, 10: 123-137.

[31] HILBERT D. Über die stetige Abbildung einer Linie auf ein Flächenstück[M]// Dritter Band: Analysis Grundlagen der Mathematik · Physik Verschiedenes. Switzerland: Springer, 1935: 1-2.

[32] MORTON G M. A computer oriented geodetic data base and a new technique in file Sequencing[R]. Ottawa, Canada: IBM Ltd., 1966.

[33] CONNOR M, KUMAR P. Fast construction of k-nearest neighbor graphs for point clouds[J]. IEEE Transactions on Visualization and Computer Graphics, 2010, 16(4):599-608.

[34] YAO Bin, LI Feifei, KUMAR P. K nearest neighbor queries and kNN-Joins in large relational databases (almost) for free[C]// IEEE 26th International Conference on Data Engineering (ICDE 2010). Long Beach, CA, USA: IEEE, 2010:4-15.

[35] DONG Wei, MOSES C, LI Kai. Efficient k-nearest neighbor graph construction for generic similarity measures[C]// Proceedings of the 20th international conference on World wide web-WWW '11. New York, USA: ACM Press, 2011:577-586.

[36] WANG Jing, WANG Jingdong, ZENG Gang, et al. Scalable k-NN graph construction for visual descriptors[C]// IEEE Conference on Computer Vision and Pattern Recognition. Providence, RI, USA:IEEE,2012: 1106-1113.

[37] ZHANG Yanming, HUANG Kaizhu, GENG Guanggang, et al. Fast kNN graph construction with locality[M]//lOCKEEL, H, KERSTING K, NIJSSEN S, et al. Machine Learning and Knowledge Discovery in Databases. ECML PKDD 2013. Lecture Notes in Computer Science(). Berlin/Heidelberg:Springer, 2013, 8189: 660-674.

[38] CALLAHAN P B. Optimal parallel all-nearest-neighbors using the well-separated pair decomposition[C]// SFCS'93: Proceedings of the 1993 IEEE 34th Annual Foundations of Computer Science. Washington, DC, USA: IEEE, 1993: 332-340.

[39] WARASHINA T, AOYAMA K, SAWADA H, et al. Efficient K-nearest neighbor graph construction using MapReduce for large-scale data sets[J]. IEICE Transactions on Information and Systems, 2014, E97-D(12): 3142-3154.

[40] MA Hengzhao, LI Jianzhong. A True $O(nlogn)$ Algorithm for the all-k-nearest-neighbors problem[M]//LI Y, CARDEI M, HUANG Y. Combinatorial Optimization and Applications. COCOA 2019. Lecture Notes in Computer Science. Cham: Springer, 2019,11949: 362-374.

[41] MA Hengzhao, LI Jianzhong. An $O(logn)$ query time algorithm for reducing ε-NN to (c, r)-NN[J]. Theoretical Computer Science, 2020, 803(10): 178-195.

[42] CAI Zhipeng, MIAO Dongjing, LI Yingshu. Deletion propagation for multiple key preserving conjunctive queries: Approximations and complexity[C]// 35th IEEE International Conference on Data Engineering, ICDE 2019. Macao, China: IEEE, 2019: 506-517.

[43] MIAO Dongjing, CAI Zhipeng, LI Jianzhong, et al. The computation of optimal subset repairs[J]. Proc. VLDB Endow., 2020, 13(11):2061-2074.

[44] MIAO Dongjing, YU Jiguo, CAI Zhipeng. The hardness of resilience for nested aggregation query[J]. Theoretical Computer Science, 2020, 803:152-159.

[45] MIAO Dongjing, CAI Zhipeng, LI Jianzhong. On the complexity of bounded view propagation for conjunctive queries[J]. IEEE Transactions on Knowledge and Data Engineering, 2018, 30(1): 115-127.

[46] MIAO Dongjing, LI Jianzhong, CAI Zhipeng. Maximum reachability preserved graph cut[J]. Theoretical Computer Science, 2020, 840:187-198.

[47] BENTLEY J L. Multidimensional binary search trees used for associative searching[J]. Communications of the ACM, 1975, 18(9): 509-517.

[48] YIANILOS P N. Data structures and algorithms for nearest neighbor search in general metric spaces[C]// SODA'93: Proceedings of the Fourth Annual ACM-SIAM Symposium on Discrete Algorithms. Austin, Texas, USA: Society for Industrial and Applied Mathematics, 1993:311-321.

[49] MUJA M, LOWE D G. Scalable nearest neighbor algorithms for high dimensional data[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2014, 36(11): 2227-2240.

[50] LIN Pengcheng, ZHAO Wanlei. Graph based nearest neighbor search: Promises and failures[J]. arXiv preprint arXiv: 1904.02077,2019.

[51] AMATO G, GENNARO C, SAVINO P. MI-File: using inverted files for scalable approximate similarity search[J]. Multimedia Tools and Applications, 2014, 71(3): 1333-1362.

[52] ESULI A. Use of permutation prefixes for efficient and scalable approximate similarity search[J]. Information Processing & Management, 2012, 48(5): 889-902.

[53] DATAR M, IMMORLICA N, INDYK P, et al. Locality-sensitive hashing scheme based on p-stable distributions[C]// Proceedings of the twentieth annual symposium on Computational geometry-SCG'04. New York, USA: ACM Press, 2004: 253-262.

[54] GONG L, WANG H, OGIHARA M, et al. iDEC[J]. Proceedings of the VLDB Endowment, 2020, 13(9): 1483-1497.

[55] LU K, WANG H, WANG W, et al. VHP[J]. Proceedings of the VLDB Endowment, 2020, 13(9):1443-1455.

[56] ANDONI A, LAARHOVEN T, RAZENSHTEYN I, et al. Optimal hashing-based time-space trade-offs for approximate near

neighbors [C]// Proceedings of the Twenty - Eighth Annual ACM-SIAM Symposium on Discrete Algorithms. Philadelphia, PA: Society for Industrial and Applied Mathematics, 2017:47-66.
[57] ANDONI A, RAZENSHTEYN I. Optimal data-dependent hashing for approximate near neighbors [C]// Proceedings of the Forty - Seventh Annual ACM on Symposium on Theory of Computing - STOC'15. New York, USA: ACM Press, 2015:793-801.
[58] GAO Jinyang, JAGADISH H, OOI B C, et al. Selective hashing: Closing the gap between radius search and K-NN search[C]// Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining - KDD'15. New York, USA: ACM Press, 2015:349-358.
[59] WANG Jingdong, SHEN Hengtao, SONG Jingkuan, et al. Hashing for similarity search: A survey[J]. arXiv preprint arXiv: 1408.2927, 2014.
[60] KRAUTHGAMER R, LEE J R. Navigating nets: Simple algorithms for proximity search[C]//Proceedings of the fifteenth annual ACM - SIAM Symposium on Discrete Algorithms. New York: ACM Press, 2004:798-807.
[61] GAN Junhao, FENG Jianlin, FANG Qiong, et al. Locality - sensitive hashing scheme based on dynamic collision counting [C]// Proceedings of the 2012 International Conference on Management of Data - SIGMOD'12. New York, USA: ACM Press, 2012:541-552.
[62] SUN Y, WANG W, QIN J, et al. SRS[J]. Proceedings of the VLDB Endowment, 2014, 8(1):1-12.
[63] MITCHELL J S, MULZER W. Proximity algorithms [M]// GOODMAN J E, O'ROURKE J, TÓTH C D. Handbook of discrete and computational geometry. New York: Chapman and Hall/CRC, 2017:1-26.
[64] PAREDES R, CHÁVEZ E. Using the k-nearest neighbor graph for proximity searching in metric spaces [M]//CONSENS M, NAVARRO G. String processing and information retrieval. SPIRE 2005. Lecture Notes in Computer Science. Berlin/ Heidelberg: Springer, 2005,3772:127-138.
[65] OCSA A, BEDREGAL C, CUADROS-VARGAS E, et al. A new approach for similarity queries using proximity graphs[C]// Simpósio Brasileiro de Banco de Dados. Brasil: dblp, 2007:131-142.
[66] MALKOV Y, PONOMARENKO A, LOGVINOV A, et al. Approximate nearest neighbor algorithm based on navigable small world graphs[J]. Information Systems, 2014, 45:61-68.
[67] MA Hengzhao, LI Jianzhong. A sub-linear time algorithm for approximating k-nearest-neighbor with full quality guarantee [M]//WU W, ZHANG Z. Com-binatorial Optimization and Applications. COCOA 2020. Lecture Notes in Computer Science. Cham: Springer, 2020, 12577:19-31.
[68] TRAD M R, JOLY A, BOUJEMAA N. Distributed KNN-graph approximation via hashing [C]// Proceedings of the 2nd ACM International Conference on Multimedia Retrieval-ICMR'12. New York, USA: ACM Press, 2012: 43.
[69] KOMAROV I, DASHTI A, D'SOUZA R M. Fast k-NNG construction with GPU-based quick multi-select[J]. Plos One, 2014, 9(5): E92409.
[70] BENTLEY J L. Multidimensional divide-and-conquer [J]. Communications of the ACM, 1980, 23(4): 214-229.
[71] VAIDYA P M. An O(n logn) algorithm for the all-nearest-neighbors problem [J]. Discrete & Computational Geometry, 1989, 4(2): 101-115.
[72] GOEL A, INDYK P, VARADARAJAN K. Reductions among high dimensional proximity problems [C]// Proceedings of the Twelfth Annual ACM-SIAM Symposium on Discrete Algorithms. Washington D.C. USA: SIAM, 2001:769-778.
[73] ZHENG Yuxin, GUO Qi, TUNG A K H, et al. LazyLSH: Approximate nearest neighbor search for multiple distance functions with a single index[C]// Proceedings of the 2016 International Conference on Management of Data-SIGMOD '16. New York, USA: ACM Press, 2016: 2023-2037.
[74] ISCEN A, TOLIAS G, AVRITHIS Y, et al. Mining on manifolds: Metric learning without labels[C]// Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Salt Lake City, UT, USA: IEEE, 2018: 7642-7651.
[75] LU Xiaolu. Improving search using proximity-based statistics[C]// Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval. Santiago Chile: ACM, 2015:1065.
[76] LUO Yucen, ZHU Jun, LI Mmengxi, et al. Smooth neighbors on teacher graphs for semi-supervised learning[C]// Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Salt Lake City, UT, USA: IEEE, 2018:8896-8905.
[77] ARYA S, MOUNT D M. Approximate nearest neighbor queries in fixed dimensions[C]//Proceedings of the Fourth Annual ACM/SIGACT-SIAM Symposium on Discrete Algorithms. Austin Texas USA: SIAM, 1993: 271-280.
[78] ARYA S, MOUNT D M, NETANYAHU N S, et al. An optimal algorithm for approximate nearest neighbor searching in fixed dimensions[J]. Journal of the ACM, 1998, 45(6):891-923.
[79] BERN M. Approximate closest-point queries in high dimensions [J]. Information Processing Letters, 1993, 45(2):95-99.
[80] HAR-PELED S. A replacement for Voronoi diagrams of near linear size [C]//Proceedings 42nd IEEE Symposium on Foundations of Computer Science. Newport Beach, CA, USA: IEEE, 2001:94-103.
[81] HAR-PELED S, INDYK P, MOTWANI R. Approximate nearest neighbor: Towards removing the curse of dimensionality [J]. Theory of Computing, 2012, 8(1):321-350.
[82] ANDONI A, INDYK P. Nearest neighbors in high-dimensional spaces[C]// Handbook of Discrete and Computational Geometry. 3rd. [S.l.]: CRC Press, Inc, 2017:1135-1155.
[83] VAIDYA P M. An optimal algorithm for the all-nearest-neighbors problem [C]// Intergovernmental Panel on Climate Change. 27th Annual Symposium on Foundations of Computer Science (sfcs 1986). Cambridge: IEEE, 1986:117-122.