# Technical Perspective: Fair Near Neighbor Search via Sampling

Qin Zhang
Indiana University
700 North Woodlawn Avenue
Bloomington, IN 47408
USA
qzhangcs@indiana.edu

One of the most important functionalities of a database system is to answer queries. We are interested in the following question: If there exists more than one answer to the given query, which one should the database report? There are two apparent choices: to return all the valid answers or to return one of them. The problem with the former choice is that it is often time-prohibitive to search for all valid answers. In the latter choice, *fairness* may become an issue, since the index built for fast search may introduce bias to the query result. For example, the index may favor a certain portion of the input data (e.g., nodes near the root of a tree index) and with a higher chance, output an answer related to that portion than other portions. Such bias can sometimes lead to undesirable consequences.

Algorithmic fairness has recently received great attention in computer science, most notably in the area of machine learning where bias in training data may be transferred to the output of the algorithm (see, e.g., [2, 1]). Conceptually, fairness can be defined as "similar items should be treated similarly", but in many settings, such as complicated social-technical systems, the precise definition of fairness is not unique and can sometimes be controversial. In the setting of database queries, fairness is often easier to define, though again the definition is not necessarily unique.

The paper "Fair Near Neighbor Search Via Sampling" by Aumüller, Har-Peled, Mahabadi, Pagh, and Silvestri studied a basic problem in similarity search called *$r$-near neighbor* ($r$-NN). In this problem, given a set of input points $S$, we want to build an index such that given a query point $p$, we can output a point $q$ in $S$ such that the distance between $p$ and $q$ is no more than $r$, in a computationally efficient manner with the assistance of the index. In the *fair* version of the $r$-NN problem, we want to return a *fair* $r$-near neighbor $q$, which is a uniform random sample from the set of all points in $S$ within a distance of $r$ from $p$.

As mentioned earlier, the idea of first enumerating all $r$-near neighbors and then randomly picking one is not feasible for real-time query processing. The standard technique for solving $r$-NN in high dimensions is *locality sensitive hashing (LSH)* [3]. In this method, we hash all points in $S$ to a set of buckets. When a query comes, we look at points that have been hashed to the bucket that contains the query point and then try to find the answer among those points. We often repeat this process multiple times in parallel to boost the success probability of finding a $r$-near neighbor (if it exists). Unfortunately, the standard LSH favors close points, and the simple idea of reporting a random point from a random bucket containing the query point will not produce a "fair" solution.

The authors approach the fair $r$-NN problem by looking at a more generic data structure problem taking a set of sets as input. Now given a subset $\mathcal{G}$ of the input sets, we want to sample an item from $\bigcup \mathcal{G}$ uniformly at random efficiently. For the $r$-NN problem, $\mathcal{G}$ corresponds to the set of buckets to which the query point is hashed. One subtlety is that there may exist false positives in these buckets due to the properties of LSH. We say a point is a false positive if it is *not* a valid $r$-near neighbor. To handle this issue, the authors augment the generic data structure by taking into consideration a marked set of outliers $\mathcal{O}$ and updating the goal to sample uniformly at random from $\bigcup \mathcal{G} \backslash \mathcal{O}$ efficiently. This is not an easy task; the solution in the paper employs several clever technical ideas.

The paper has the potential for high impact in the emerging area of fair data structure and database system design. Despite its importance, limited research has been done in this frontier. This work can serve as a catalyst along this line of research in the years to come.

## 1. REFERENCES

[1] Sam Corbett-Davies, Emma Pierson, Avi Feller, Sharad Goel, and Aziz Huq. Algorithmic decision making and the cost of fairness. In *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 797–806, 2017.

[2] Moritz Hardt, Eric Price, and Nati Srebro. Equality of opportunity in supervised learning. In *Advances in Neural Information Processing Systems*, pages 3315–3323, 2016.

[3] Piotr Indyk and Rajeev Motwani. Approximate nearest neighbors: Towards removing the curse of dimensionality. In *ACM Symposium on Theory of Computing*, pages 604–613, 1998.