Fast k-Nearest Neighbour Search via Prioritized DCI

Supplementary Material

Ke Li [1] **Jitendra Malik** [1]

7. Analysis

We present proofs that were omitted from the main paper below.

Theorem 1. *Let $\left\{v_i^l\right\}_{i=1}^N$ and $\left\{v_{i'}^s\right\}_{i'=1}^{N'}$ be sets of vectors such that $\left\|v_i^l\right\|_2 > \left\|v_{i'}^s\right\|_2 \quad \forall i \in [N], i' \in [N']$. Furthermore, let $\left\{u'_{ij}\right\}_{i\in[N],j\in[M]}$ be random uniformly distributed unit vectors such that $u'_{i1}, \ldots, u'_{iM}$ are independent for any given i. Consider the events $\left\{\exists v_{i'}^s \text{ s.t. } \max_j \left\{\left|\langle v_i^l, u'_{ij}\rangle\right|\right\} \leq \left\|v_{i'}^s\right\|_2\right\}_{i=1}^N$. The probability that at least k' of these events occur is at most $\frac{1}{k'}\sum_{i=1}^N \left(1 - \frac{2}{\pi}\cos^{-1}\left(\left\|v_{\max}^s\right\|_2 / \left\|v_i^l\right\|_2\right)\right)^M$, where $\left\|v_{\max}^s\right\|_2 = \max_{i'}\left\{\left\|v_{i'}^s\right\|_2\right\}$. Furthermore, if $k' = N$, it is at most $\min_{i\in[N]}\left\{\left(1 - \frac{2}{\pi}\cos^{-1}\left(\left\|v_{\max}^s\right\|_2 / \left\|v_i^l\right\|_2\right)\right)^M\right\}$.*

Proof. The event that $\exists v_{i'}^s$ s.t. $\max_j \left\{\left|\langle v_i^l, u'_{ij}\rangle\right|\right\} \leq \left\|v_{i'}^s\right\|_2$ is equivalent to the event that $\max_j \left\{\left|\langle v_i^l, u'_{ij}\rangle\right|\right\} \leq \max_{i'}\left\{\left\|v_{i'}^s\right\|_2\right\} = \left\|v_{\max}^s\right\|_2$. Take E_i to be the event that $\max_j \left\{\left|\langle v_i^l, u'_{ij}\rangle\right|\right\} \leq \left\|v_{\max}^s\right\|_2$. By Lemma 1, $\Pr(E_i) \leq \left(1 - \frac{2}{\pi}\cos^{-1}\left(\left\|v_{\max}^s\right\|_2 / \left\|v_i^l\right\|_2\right)\right)^M$. It follows from Lemma 2 that the probability that k' of E_i's occur is at most $\frac{1}{k'}\sum_{i=1}^N \Pr(E_i) \leq \frac{1}{k'}\sum_{i=1}^N \left(1 - \frac{2}{\pi}\cos^{-1}\left(\left\|v_{\max}^s\right\|_2 / \left\|v_i^l\right\|_2\right)\right)^M$. If $k' = N$, we use the fact that $\bigcap_{i'=1}^N E_{i'} \subseteq E_i \quad \forall i$, which implies that $\Pr\left(\bigcap_{i'=1}^N E_{i'}\right) \leq \min_{i\in[N]}\Pr(E_i) \leq \min_{i\in[N]}\left\{\left(1 - \frac{2}{\pi}\cos^{-1}\left(\left\|v_{\max}^s\right\|_2 / \left\|v_i^l\right\|_2\right)\right)^M\right\}$. □

Lemma 3. *Consider points in the order they are retrieved from a composite index that consists of m simple indices. The probability that there are at least n_0 points that are not the true k-nearest neighbours but are retrieved before some of them is at most $\frac{1}{n_0-k}\sum_{i=2k+1}^n \left(1 - \frac{2}{\pi}\cos^{-1}\left(\left\|p^{(k)} - q\right\|_2 / \left\|p^{(i)} - q\right\|_2\right)\right)^m$.*

Proof. Points that are not the true k-nearest neighbours but are retrieved before some of them will be referred to as *extraneous points* and are divided into two categories: *reasonable* and *silly*. An extraneous point is reasonable if it is one of the $2k$-nearest neighbours, and is silly otherwise. For there to be n_0 extraneous points, there must be $n_0 - k$ silly extraneous points. Therefore, the probability that there are n_0 extraneous points is upper bounded by the probability that there are $n_0 - k$ silly extraneous points.

Since points are retrieved from the composite index in the order of increasing maximum projected distance to the query, for any pair of points p and p', if p is retrieved before p', then $\max_j\left\{\left|\langle p - q, u_{jl}\rangle\right|\right\} \leq \max_j\left\{\left|\langle p' - q, u_{jl}\rangle\right|\right\}$, where $\left\{u_{jl}\right\}_{j=1}^m$ are the projection directions associated with the constituent simple indices of the composite index.

By Theorem 1, if we take $\left\{v_i^l\right\}_{i=1}^N$ to be $\left\{p^{(i)} - q\right\}_{i=2k+1}^n$, $\left\{v_{i'}^s\right\}_{i'=1}^{N'}$ to be $\left\{p^{(i)} - q\right\}_{i=1}^k$, M to be m, $\left\{u'_{ij}\right\}_{j\in[M]}$ to be $\left\{u_{jl}\right\}_{j\in[m]}$ for all $i \in [N]$ and k' to be $n_0 - k$, we obtain an upper bound for the probability of there being a subset of $\left\{p^{(i)}\right\}_{i=2k+1}^n$ of size $n_0 - k$ such that for all points p in the subset, $\max_j\left\{\left|\langle p - q, u_{jl}\rangle\right|\right\} \leq \left\|p' - q\right\|_2$ for some $p' \in \left\{p^{(i)} - q\right\}_{i=1}^k$. In other words, this is the probability of there being $n_0 - k$ points that are not the $2k$-nearest neighbours whose maximum projected distances are no greater than the distance from some k-nearest neighbours to the query, which is at most $\frac{1}{n_0-k}\sum_{i=2k+1}^n \left(1 - \frac{2}{\pi}\cos^{-1}\left(\left\|p^{(k)} - q\right\|_2 / \left\|p^{(i)} - q\right\|_2\right)\right)^m$. Since the event that $\max_j\left\{\left|\langle p - q, u_{jl}\rangle\right|\right\} \leq \max_j\left\{\left|\langle p' - q, u_{jl}\rangle\right|\right\}$ is contained in the event that $\max_j\left\{\left|\langle p - q, u_{jl}\rangle\right|\right\} \leq \left\|p' - q\right\|_2$ for any p, p', this is also an upper bound for the probability of there being $n_0 - k$ points that are not the $2k$-nearest neighbours whose maximum projected distances do not exceed those of some of the k-nearest neighbours, which by definition is the probability that there are $n_0 - k$ silly extraneous points. Since this probability is no less than the probability that there are n_0 extraneous points, the upper bound also applies to this probability. □

Lemma 4. *Consider point projections in a composite index that consists of m simple indices in the order they are visited. The probability that n_0 point projections that are not of the true k-nearest neighbours are visited before all true k-nearest neighbours have been retrieved is at most $\frac{m}{n_0-mk}\sum_{i=2k+1}^n \left(1 - \frac{2}{\pi}\cos^{-1}\left(\left\|p^{(k)} - q\right\|_2 / \left\|p^{(i)} - q\right\|_2\right)\right)$.*

Proof. Projections of points that are not the true k-nearest neighbours but are visited before the k-nearest neighbours have all been retrieved will be referred to as *extraneous projections* and are divided into two categories: *reasonable* and *silly*. An extraneous projection is reasonable if it is of one of the $2k$-nearest neighbours, and is silly otherwise. For there to be n_0 extraneous projections, there must be $n_0 - mk$ silly extraneous projections, since there could be at most mk reasonable extraneous projections. Therefore, the probability that there are n_0 extraneous projections is upper bounded by the probability that there are $n_0 - mk$ silly extraneous projections.

Since point projections are visited in the order of increasing projected distance to the query, each extraneous silly projection must be closer to the query projection than the maximum projection of some k-nearest neighbour.

By Theorem 1, if we take $\left\{v_i^l\right\}_{i=1}^N$ to be $\left\{p^{(2k+\lfloor(i-1)/m\rfloor+1)} - q\right\}_{i=1}^{m(n-2k)}$, $\left\{v_{i'}^s\right\}_{i'=1}^{N'}$ to be $\left\{p^{(\lfloor(i-1)/m\rfloor+1)} - q\right\}_{i=1}^{mk}$, M to be 1, $\left\{u_{i1}'\right\}_{i=1}^N$ to be $\left\{u_{(i \mod m),l}\right\}_{i=1}^{m(n-2k)}$ and k' to be $n_0 - mk$, we obtain an upper bound for the probability of there being $n_0 - mk$ point projections that are not of the $2k$-nearest neighbours whose distances to their respective query projections are no greater than the true distance between the query and some k-nearest neighbour, which is $\frac{1}{n_0-mk}\sum_{i=2k+1}^n m\left(1 - \frac{2}{\pi}\cos^{-1}\left(\frac{\left\|p^{(k)}-q\right\|_2}{\left\|p^{(i)}-q\right\|_2}\right)\right)$.

Because maximum projected distances are no more than true distances, this is also an upper bound for the probability of there being $n_0 - mk$ silly extraneous projections. Since this probability is no less than the probability that there are n_0 extraneous projections, the upper bound also applies to this probability. □

Lemma 5. *On a dataset with global relative sparsity (k, γ), the quantity $\sum_{i=2k+1}^n \left(1 - \frac{2}{\pi}\cos^{-1}\left(\left\|p^{(k)} - q\right\|_2 / \left\|p^{(i)} - q\right\|_2\right)\right)^m$ is at most $O\left(k\max(\log(n/k), (n/k)^{1-m\log_2\gamma})\right)$.*

Proof. By definition of global relative sparsity, for all $i \geq 2k+1$, $\left\|p^{(i)} - q\right\|_2 > \gamma\left\|p^{(k)} - q\right\|_2$. A recursive application shows that for all $i \geq 2^{i'}k + 1$, $\left\|p^{(i)} - q\right\|_2 > \gamma^{i'}\left\|p^{(k)} - q\right\|_2$.

Applying the fact that $1 - (2/\pi)\cos^{-1}(x) \leq x\ \forall x \in [0, 1]$ and the above observation yields:

$$\sum_{i=2k+1}^n \left(1 - \frac{2}{\pi}\cos^{-1}\left(\frac{\left\|p^{(k)} - q\right\|_2}{\left\|p^{(i)} - q\right\|_2}\right)\right)^m$$
$$\leq \sum_{i=2k+1}^n \left(\frac{\left\|p^{(k)} - q\right\|_2}{\left\|p^{(i)} - q\right\|_2}\right)^m$$
$$< \sum_{i'=1}^{\lceil\log_2(n/k)\rceil-1} 2^{i'}k\gamma^{-i'm}$$

If $\gamma \geq \sqrt[m]{2}$, this quantity is at most $k\log_2(n/k)$. On the other hand, if $1 \leq \gamma < \sqrt[m]{2}$, this quantity can be simplified to:

$$k\left(\frac{2}{\gamma^m}\right)\left(\left(\frac{2}{\gamma^m}\right)^{\lceil\log_2(n/k)\rceil-1} - 1\right) / \left(\frac{2}{\gamma^m} - 1\right)$$
$$= O\left(k\left(\frac{2}{\gamma^m}\right)^{\lceil\log_2(n/k)\rceil-1}\right)$$
$$= O\left(k\left(\frac{n}{k}\right)^{1-m\log_2\gamma}\right)$$

Therefore, $\sum_{i=2k+1}^n \left(\left\|p^{(k)} - q\right\|_2 / \left\|p^{(i)} - q\right\|_2\right)^m \leq O\left(k\max(\log(n/k), (n/k)^{1-m\log_2\gamma})\right)$. □

Lemma 6. *For a dataset with global relative sparsity (k, γ) and a given composite index consisting of m simple indices, there is some $k_0 \in \Omega(k\max(\log(n/k), (n/k)^{1-m\log_2\gamma}))$ such that the probability that the candidate points retrieved from the composite index do not include some of the true k-nearest neighbours is at most some constant $\alpha_0 < 1$.*

Proof. We will refer to the true k-nearest neighbours that are among first k_0 points retrieved from the composite index as *true positives* and those that are not as *false negatives*. Additionally, we will refer to points that are not true k-nearest neighbours but are among the first k_0 points retrieved as *false positives*.

When not all the true k-nearest neighbours are among the first k_0 candidate points, there must be at least one false negative and so there can be at most $k-1$ true positives. Consequently, there must be at least $k_0 - (k-1)$ false positives. To find an upper bound on the probability of the existence of $k_0 - (k-1)$ false positives in terms of global relative sparsity, we apply Lemma 3 with n_0 set to $k_0 - (k-1)$, followed by Lemma 5. We conclude that this probability is at most $\frac{1}{k_0-2k+1}O\left(k\max(\log(n/k), (n/k)^{1-m\log_2\gamma})\right)$. Because the event that not all the true k-nearest neighbours are among the first k_0 candidate points is contained in the event that there are $k_0 - (k-1)$ false positives, the former is upper bounded by the same quantity. So, we can choose some $k_0 \in \Omega(k\max(\log(n/k), (n/k)^{1-m\log_2\gamma}))$ to make it strictly less than 1. □

Lemma 7. *For a dataset with global relative sparsity (k, γ) and a given composite index consisting of m simple indices, there is some $k_1 \in$*

$\Omega(mk\max(\log(n/k),(n/k)^{1-\log_2\gamma}))$ *such that the probability that the candidate points retrieved from the composite index do not include some of the true k-nearest neighbours is at most some constant $\alpha_1 < 1$.*

Proof. We will refer to the projections of true k-nearest neighbours that are among first k_1 visited point projections as *true positives* and those that are not as *false negatives*. Additionally, we will refer to projections of points that are not of the true k-nearest neighbours but are among the first k_1 visited point projections as *false positives*.

When a k-nearest neighbour is not among the candidate points that have been retrieved, some of its projections must not be among the first k_1 visited point projections. So, there must be at least one false negative, implying that there can be at most $mk-1$ true positives. Consequently, there must be at least $k_1-(mk-1)$ false positives. To find an upper bound on the probability of the existence of $k_1-(mk-1)$ false positives in terms of global relative sparsity, we apply Lemma 4 with n_0 set to $k_1-(mk-1)$, followed by Lemma 5. We conclude that this probability is at most $\frac{m}{k_1-2mk+1}O\left(k\max(\log(n/k),(n/k)^{1-\log_2\gamma})\right)$. Because the event that some true k-nearest neighbour is missing from the candidate points is contained in the event that there are $k_1-(mk-1)$ false positives, the former is upper bounded by the same quantity. So, we can choose some $k_1 \in \Omega(mk\max(\log(n/k),(n/k)^{1-\log_2\gamma}))$ to make it strictly less than 1. □

Theorem 2. *For a dataset with global relative sparsity (k,γ), for any $\epsilon > 0$, there is some L, $k_0 \in \Omega(k\max(\log(n/k),(n/k)^{1-m\log_2\gamma}))$ and $k_1 \in \Omega(mk\max(\log(n/k),(n/k)^{1-\log_2\gamma}))$ such that the algorithm returns the correct set of k-nearest neighbours with probability of at least $1-\epsilon$.*

Proof. For a given composite index, by Lemma 6, there is some $k_0 \in \Omega(k\max(\log(n/k),(n/k)^{1-m\log_2\gamma}))$ such that the probability that some of the true k-nearest neighbours are missed is at most some constant $\alpha_0 < 1$. Likewise, by Lemma 7, there is some $k_1 \in \Omega(mk\max(\log(n/k),(n/k)^{1-\log_2\gamma}))$ such that this probability is at most some constant $\alpha_1 < 1$. By choosing such k_0 and k_1, this probability is therefore at most $\min\{\alpha_0,\alpha_1\} < 1$. For the algorithm to fail, all composite indices must miss some k-nearest neighbours. Since each composite index is constructed independently, the algorithm fails with probability of at most $(\min\{\alpha_0,\alpha_1\})^L$, and so must succeed with probability of at least $1-(\min\{\alpha_0,\alpha_1\})^L$. Since $\min\{\alpha_0,\alpha_1\} < 1$, there is some L that makes $1-(\min\{\alpha_0,\alpha_1\})^L \geq 1-\epsilon$. □

Theorem 3. *For a given number of simple indices m, the algorithm takes $O\left(dk\max(\log(n/k),(n/k)^{1-m/d'})+mk\log m\left(\max(\log(n/k),(n/k)^{1-1/d'})\right)\right)$ time to retrieve the k-nearest neighbours at query time, where d' denotes the intrinsic dimensionality.*

Proof. Computing projections of the query point along all u_{jl}'s takes $O(dm)$ time, since L is a constant. Searching in the binary search trees/skip lists T_{jl}'s takes $O(m\log n)$ time. The total number of point projections that are visited is at most $\Theta(mk\max(\log(n/k),(n/k)^{1-\log_2\gamma}))$. Because determining the next point to visit requires popping and pushing a priority queue, which takes $O(\log m)$ time, the total time spent on visiting points is $O(mk\log m\max(\log(n/k),(n/k)^{1-\log_2\gamma}))$. The total number of candidate points retrieved is at most $\Theta(k\max(\log(n/k),(n/k)^{1-m\log_2\gamma}))$. Because true distances are computed for every candidate point, the total time spent on distance computation is $O(dk\max(\log(n/k),(n/k)^{1-m\log_2\gamma}))$. We can find the k closest points to the query among the candidate points using a selection algorithm like quickselect, which takes $O(k\max(\log(n/k),(n/k)^{1-m\log_2\gamma}))$ time on average. Since the time for visiting points and for computing distances dominates, the entire algorithm takes $O(dk\max(\log(n/k),(n/k)^{1-m\log_2\gamma}) + mk\log m\max(\log(n/k),(n/k)^{1-\log_2\gamma}))$ time. Substituting $1/d'$ for $\log_2\gamma$ yields the desired expression. □

Theorem 4. *For a given number of simple indices m, the algorithm takes $O(m(dn+n\log n))$ time to preprocess the data points in D at construction time.*

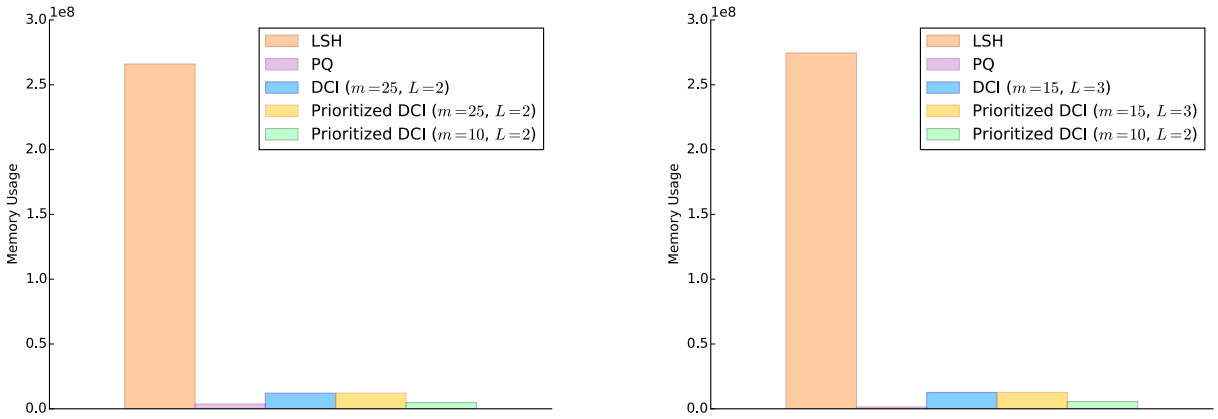
Proof. Computing projections of all n points along all u_{jl}'s takes $O(dmn)$ time, since L is a constant. Inserting all n points into mL self-balancing binary search trees/skip lists takes $O(mn\log n)$ time. □

Theorem 5. *The algorithm requires $O(m(d+\log n))$ time to insert a new data point and $O(m\log n)$ time to delete a data point.*

Proof. In order to insert a data point, we need to compute its projection along all u_{jl}'s and insert it into each binary search tree or skip list. Computing the projections takes $O(md)$ time and inserting them into the corresponding self-balancing binary search trees or skip lists takes $O(m\log n)$ time. In order to delete a data point, we simply remove its projections from each of the binary search trees or skip lists, which takes $O(m\log n)$ time. □

Theorem 6. *The algorithm requires $O(mn)$ space in addition to the space used to store the data.*

Proof. The only additional information that needs to be stored are the mL binary search trees or skip lists. Since



(a) (b)

Figure 3. Memory usage of different algorithms on (a) CIFAR-100 and (b) MNIST. Lower values are better.

n entries are stored in each binary search tree/skip list, the total additional space required is $O(mn)$. □

8. Experiments

Figure 3 shows the memory usage of different algorithms on CIFAR-100 and MNIST.