# Locality sensitive hashing and semantic hashing

Tianpei Xie

Sep. 3rd., 2022

## Contents

# 1 Definitions of locality sensitive hashing

- Given a dataset $\mathcal{D}$ with $n$ points and $d$ dimensions and a query point $q$ in the same space as the dataset, the *goal* of ***c-ANN search*** (where $c = (1+\epsilon) > 1$ is an approximation ratio) is to return points $o \in \mathcal{D}$ such that $\text{dist}(o, q) \leq c \times \text{dist}(o^*, q)$, where $o^*$ is the true nearest neighbor of $q$ in $\mathcal{D}$ and dist is the distance between the two points. Similarily, ***c-k-ANN search*** aims at returning top-$k$ points such that $\text{dist}(o_i, q) \leq c \times \text{dist}(o_i^*, q)$, where $1 \leq i \leq k$. [Jafari et al., 2021] $c$-ANN is also called **$\epsilon$-ANN** given $c = (1+\epsilon) > 1$.
- Hashing-based methods try to find the nearest neighbors in high-dimensional datasets by **projecting** them into one or more low-dimensional spaces using *hash functions*. **Locality sensitive hashing (LSH)** is a famous hashing-based method that creates the lowd-imensional projections such that the *localities of the original space are preserved* in them (i.e. two nearby points in the original space are also nearby in the projected space).
- For two points $\boldsymbol{x}$ and $\boldsymbol{y}$ in a $d$-dimensional dataset $\mathcal{D} \subset \mathbb{R}^d$, we say a ***hash function*** $H$ is $(R, cR, p_1, p_2)$**-sensitive** if it satisfies the following two conditions:

  1. if $|\boldsymbol{x} - \boldsymbol{y}| \leq R$, then $\mathbb{P}[H(\boldsymbol{x}) = H(\boldsymbol{y})] \geq p_1$, and
  2. if $|\boldsymbol{x} - \boldsymbol{y}| > cR$, then $\mathbb{P}[H(\boldsymbol{x}) = H(\boldsymbol{y})] \leq p_2$

  Here, $c$ is an approximation ratio and $p_1$ and $p_2$ are probabilities. In order for this definition to work, $c > 1$ and $p_1 > p_2$.

  The definition states that two points $\boldsymbol{x}$ and $\boldsymbol{y}$ are hashed to the **same bucket** in the projection with a ***very high probability*** $\geq p_1$ if they **are close to each other**, and if they are **not close to each other**, then they will be hashed to the same bucket with a ***low probability*** $\leq p_2$. Next, we present the popular hash function families for the Hamming, Minkowski, Angular, and Jaccard distances.

# 2 Hamming-based Locality Hashing

*Locality Sensitive Hashing* was first proposed in [Indyk and Motwani, 1998] for the Hamming distance to solve the $(R, c)$**-near neighbor search problem**. The proposed method uses multiple hash functions and hash tables to be able to guarantee a good search quality. Moreover, authors theoretically find the optimal number of hash functions and hash tables in order to have constant hashing probabilities.

The **Hamming distance** for comparing two **binary** data strings of equal length is the number of positions at which the corresponding symbols are different.

$$\text{dist}_H(\boldsymbol{x}, \boldsymbol{y}) = \sum_i^d |x_i - y_i|$$

where $x_i, y_i \in \{0, 1\}$ for all $i \in [1, d]$. [Indyk and Motwani, 1998] defined the LSH function as

$$H(\boldsymbol{x}) = x_i, \tag{1}$$

where $x_i$ is the $i$-th dimension of the point $\boldsymbol{x}$ for some $i \in [1, d]$. Therefore, for two points $\boldsymbol{x}$ and $\boldsymbol{y}$ with a ***Hamming distance*** of $r$, the probability that they have the same hash value is

$$\mathbb{P}[H(\boldsymbol{x}) = H(\boldsymbol{y})] = 1 - \frac{r}{d}$$

. This LSH function is called ***bit sampling***. A *random* function $H$ simply selects ***a random bit*** from the input point.

## 3 Jaccard-based Locality Hashing

The **Jaccard distance** between two sets $A$ and $B$ is defined as

$$\text{dist}_J(A, B) = \frac{|A \cap B|}{|A \cup B|} \tag{2}$$

It is also called ***set resemblance*** in [Indyk and Motwani, 1998, Broder et al., 2000].

Assume that for all documents of interest $S \subset [1, \ldots, n]$. The LSH functions that preserve the **Jaccard distance** [Broder et al., 2000] is defined as

$$H(A) = \min_{x_i \in A} \{\pi(x_i)\}, \tag{3}$$

where $x_i \in A \subset S$ and $\pi$ is a ***random permutation*** on the index set $S$ from the set of *all possible permutations* $\Pi$. Therefore, for two points $A$ and $B$ with a Jaccard similarity of $J$, the probability that they have the same hash value is

$$\mathbb{P}[H(A) = H(B)] = J$$

.

Define the function family $\mathcal{H}$ to be the set of all such functions and let $D$ be the uniform distribution. Given two sets $A, B \subset S$, and $H(A) = H(B)$, we need to show that the minimizer in $A \cup B$, $x_s := \arg\min_{x_i \in A \cup B}\{\pi(x_i)\}$ lies in $A \cap B$, i.e. $x_s \in A \cap B$

$$\begin{aligned}
x_s &:= \arg \min_{x_i \in A \cup B} \{\pi(x_i)\} \\
\text{since } \min_{x_i \in A} \{\pi(x_i)\} &= \min_{x_i \in B} \{\pi(x_i)\} \\
\Rightarrow r = \pi(x_s) &= \min_{x_i \in A} \{\pi(x_i)\} = \min_{x_i \in B} \{\pi(x_i)\} \\
\Rightarrow r &= \min_{x_i \in A \cap B} \{\pi(x_i)\} \Rightarrow x_s = \arg \min_{x_i \in A \cap B} \{\pi(x_i)\}
\end{aligned}$$

As $H$ was chosen uniformly at random, $\mathbb{P}[H(A) = H(B)] = \mathbb{P}\{x_s = \arg\min_{x_i \in A \cup B}\{\pi(x_i)\} : x_s \in A \cap B, \pi \in D\} = \mathbb{P}\{A \cap B | A \cup B\} = J$.

In practice, as in the case of hashing discussed in [Broder et al., 2000], we have to deal with the sad reality that it is impossible to choose $\pi$ uniformly at random in $\Pi$. We are thus led to consider smaller families of permutations that still satisfy the ***min-wise independence condition***.

Let $\Pi$ be the set of all permutations of $[n]$. We say that *a family of permutations* $\mathcal{F} \subset \Pi$ is ***pair-wise independent*** if for any $\{x_1, x_2, y_1, y_2\} \subset [n]$ with $x_1 \neq x_2$ and $y_1 \neq y_2$,

$$\mathbb{P}\{\pi(x_1) = y_1 \wedge \pi(x_2) = y_2\} = \frac{1}{n(n-1)}$$

In a similar vein, in this paper, we say that a family of permutations $\mathcal{F} \subset \Pi$ is exactly ***min-wise independent*** (or just min-wise independent where the meaning is clear) if for any set $A \subset [n]$ and any $x \in A$, when $\pi$ is chosen at random in $\mathcal{F}$ we have

$$\mathbb{P}\left\{\min\left\{\pi(A)\right\} = \pi(x)\right\} = \frac{1}{|A|}.$$

We say that $\mathcal{F} \subset \Pi$ is ***k-restricted min-wise independent*** (or just restricted min-wise independent where the meaning is clear) if for any set $A \subseteq [n]$ with $|A| \leq k$ and any$x \in A$, when $\pi$ is chosen at random in $\mathcal{F}$ we have

$$\mathbb{P}\left\{\min\left\{\pi(A)\right\} = \pi(x)\right\} = \frac{1}{|A|}, \quad |A| \leq k.$$

# 4 Angular-based Locality Hashing

Euclidean distance on a sphere corresponds to the *angular distance* or ***cosine similarity***. For $\boldsymbol{x}, \boldsymbol{y} \in \mathcal{S}_d \subset \mathbb{R}^d$, i.e. $\|\boldsymbol{x}\|_2 = \|\boldsymbol{y}\|_2 = 1$. The ***cosine similarity*** is defined as

$$\cos(\theta_{x,y}) = \frac{\langle \boldsymbol{x}\,,\,\boldsymbol{y}\rangle}{\|\boldsymbol{x}\|_2\,\|\boldsymbol{y}\|_2} \tag{4}$$

And the normalized angle, referred to as ***angular distance***, between any two vectors $\boldsymbol{x}, \boldsymbol{y} \in \mathcal{S}_d \subset \mathbb{R}^d$ is a formal *distance metric* and can be calculated from the cosine similarity.

$$\text{dist}_\theta(\boldsymbol{x}, \boldsymbol{y}) = \frac{\theta_{x,y}}{\pi} = \frac{\arccos\left(\langle \boldsymbol{x}\,,\,\boldsymbol{y}\rangle\right)}{\pi} \tag{5}$$

For the **angular metric**, [Chávez et al., 2001] defined the LSH functions as

$$H(\boldsymbol{x}) = \text{sgn}\{\langle \boldsymbol{a}\,,\,\boldsymbol{x}\rangle\}, \tag{6}$$

where $\text{sgn}(\cdot) \in \{-1, 1\}$ is the ***sign function*** and $\boldsymbol{a}$ is a *random vector* drawn from the **Normal distribution**. In this case, for two points $\boldsymbol{x}$ and $\boldsymbol{y}$ with $\theta_{x,y}$ defined as the angle between them, the probability that they have the same hash value is

$$\mathbb{P}[H(\boldsymbol{x}) = H(\boldsymbol{y})] = 1 - \frac{\theta_{x,y}}{d}$$

.

# 5 Minkowski-based LSH Techniques

The $\ell_p$ norm is

$$\|\boldsymbol{x}\|_p = \left(\sum_i |x|^p\right)^{\frac{1}{p}}$$

A distribution $D$ over $\mathcal{R}$ is called ***p-stable***, if there exists $p \geq 0$ such that for any $n$ real numbers $v_1, \ldots, v_n$ and i.i.d. variables $X_1, \ldots, X_n$ with distribution $D$, the random variable $\sum_i v_i X_i$ has **the same distribution** as the variable $\|\boldsymbol{v}\|_p\, X = (\sum_i |x|^p)^{\frac{1}{p}} X$, where $X$ is a random variable with distribution $D$.

- *The Cauchy distribution* $D_C$, defined by the density function

$$f_1(x) = \frac{1}{\pi}\frac{1}{1+x^2}$$

  is **1-stable**

- *The Gaussian (normal) distribution* $D_G$, defined by the density function

$$f_2(x) = \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{x^2}{2}\right)$$

  is **2-stable**

For the $\ell_p$ norm, [Datar et al., 2004] defined the LSH functions as

$$H_{\boldsymbol{a},b}(\boldsymbol{x}) = \left\lfloor \frac{\langle \boldsymbol{a} \, , \, \boldsymbol{x} \rangle + b}{w} \right\rfloor \tag{7}$$

where $\boldsymbol{a}$ is a $d$-dimensional random vector chosen from the standard ***p-stable distribution*** and $b$ is a real number chosen uniformly from $[0, w)$, such that $w$ is the ***width*** of the hash bucket. For two points $\boldsymbol{x}$ and $\boldsymbol{y}$ with a $\ell_p$ distance of $r$, the probability that they have the same hash value is

$$\mathbb{P}[H(\boldsymbol{x}) = H(\boldsymbol{y})] = \int_0^w \frac{1}{r} f_p\left(\frac{t}{r}\right)\left(1 - \frac{t}{w}\right) dt.$$

Here, $f_p\left(t\right)$ is the density function of the $p$-stable distribution.

# References


Andrei Z Broder, Moses Charikar, Alan M Frieze, and Michael Mitzenmacher. Min-wise independent permutations. *Journal of Computer and System Sciences*, 60:630–659, 2000.

Edgar Chávez, Gonzalo Navarro, Ricardo Baeza-Yates, and José Luis Marroquín. Searching in metric spaces. *ACM computing surveys (CSUR)*, 33(3):273–321, 2001.

Mayur Datar, Nicole Immorlica, Piotr Indyk, and Vahab S Mirrokni. Locality-sensitive hashing scheme based on p-stable distributions. In *Proceedings of the twentieth annual symposium on Computational geometry*, pages 253–262, 2004.

Piotr Indyk and Rajeev Motwani. Approximate nearest neighbors: towards removing the curse of dimensionality. In *Proceedings of the thirtieth annual ACM symposium on Theory of computing*, pages 604–613, 1998.

Omid Jafari, Preeti Maurya, Parth Nagarkar, Khandker Mushfiqul Islam, and Chidambaram Crushev. A survey on locality sensitive hashing algorithms and their applications. *arXiv preprint arXiv:2102.08942*, 2021.