Asymmetric Locality Sensitive Hashing for Sublinear Time Maximum Inner Product Search

Evgeny Nikishin

06.10.2016

Problem formulation and motivation

Given a collection $X \in \mathbb{R}^{N \times D}$. For a feature vector $q \in \mathbb{R}^D$ find

$$p = \arg\max_{x \in X} \langle q, x \rangle$$

Problem formulation and motivation

Given a collection $X \in \mathbb{R}^{N \times D}$. For a feature vector $q \in \mathbb{R}^D$ find

$$p = \arg\max_{x \in X} \langle q, x \rangle$$

Applications:

- Recommender systems
- Multiclass classification
- ...

From MIPS to Near Neighbour Search

Assumptions:

- $||x|| \leq U < 1 \quad \forall x \in X$. If this is not the case then scale all vectors $x = \frac{U}{\max_{x \in X} ||x||} \times x$
- $||q|| = 1$ for simplicity. It can be easily removed

Define two vector transformations $P : \mathbb{R}^D \mapsto \mathbb{R}^{D+m}$ and $Q : \mathbb{R}^D \mapsto \mathbb{R}^{D+m}$ as follows:

$$P(x) = [x; ||x||^2; ||x||^4; \ldots ; ||x||^{2^m}] \quad Q(q) = [q; 1/2; 1/2; \ldots ; 1/2]$$

From MIPS to Near Neighbour Search

$$P(x) = [x; ||x||^2; ||x||^4; \ldots; ||x||^{2^m}] \quad Q(q) = [q; 1/2; 1/2; \ldots; 1/2]$$

By observing

$$2\langle Q(q), P(x)\rangle = 2\langle q, x\rangle + ||x||^2 + ||x||^4 + \ldots + ||x||^{2^m}$$

$$||P(x)||^2 = ||x||^2 + ||x||^4 + \ldots + ||x||^{2^m} + ||x||^{2^{m+1}}$$

we obtain key equality:

$$||Q(q) - P(x)||^2 = (1 + m/4) - 2\langle q, x\rangle + ||x||^{2^{m+1}}$$

From MIPS to Near Neighbour Search

$$P(x) = [x; ||x||^2; ||x||^4; \ldots; ||x||^{2^m}] \quad Q(q) = [q; 1/2; 1/2; \ldots; 1/2]$$

By observing

$$2\langle Q(q), P(x)\rangle = 2\langle q, x\rangle + ||x||^2 + ||x||^4 + \ldots + ||x||^{2^m}$$

$$||P(x)||^2 = ||x||^2 + ||x||^4 + \ldots + ||x||^{2^m} + ||x||^{2^{m+1}}$$

we obtain key equality:

$$||Q(q) - P(x)||^2 = (1 + m/4) - 2\langle q, x\rangle + ||x||^{2^{m+1}}$$

So

$$\arg\max_{x \in X} \langle q, x\rangle \approx \arg\min_{x \in X} ||Q(q) - P(x)||$$

Intuition for following definition: similar objects are desired to have equal hashes with high probability

Definition A family of hash functions $\mathcal{H}$ called (d_1, d_2, p_1, p_2)-sensitive if, for a given q and any $x \in X$,

- if $Sim(q, x) \geq d_1$ then $\mathbb{P}_{h \in \mathcal{H}}(h(Q(q)) = h(P(x))) \geq p_1$
- if $Sim(q, x) \leq d_2$ then $\mathbb{P}_{h \in \mathcal{H}}(h(Q(q)) = h(P(x))) \leq p_2$

Example: hash function for euclidean distance
Given a parameter r, we choose a random vector a with each component generated from i.i.d. standard normal, i.e. $a_i \sim \mathcal{N}(0, 1)$, and a scalar b generated uniformly from $[0, r]$

$$h(x) = \left\lfloor \frac{\langle a, x \rangle + b}{r} \right\rfloor$$

It corresponds to some line in $\mathbb{R}^D$, divided by segments with length r and returns number of segment.

ALSH for approximate NNS

Definition Data structure solves c-approximate Nearest Neighbour problem (c-NN) if, for a given parameters $d_1 > 0, \delta > 0$ and a query q, it does the following with probability $1 - \delta$: if there exists an d_1-near neighbour of q, it reports some cd_1-near neighbour of q.

Theorem Given a (d_1, cd_1, p_1, p_2)-sensitive family of hash functions, one can construct a data structure for c-NN with $O(n^\rho \log(n))$ query time, where $\rho = \frac{\log p_1}{\log p_2}$

For multiclass classification I had:

- 25% — Top 1 result
- 50% — Top 5 result
- 80% — Top 25 result

Workflow for MIPS with ALSH

1. Choose hash function h (uniformly from $\mathcal{H}$)
2. Create a hash table by applying this function to all $x \in X$, preprocessed by $P(x)$
3. For a query, compute $h(Q(q))$
4. Choose the nearest sample from hash table cell