Scalable ML

10605-10805

Hash Functions

Barnabás Póczos

Randomized Algorithms, Section 8.4

Rajeev Motwani and Prabhakar Raghavan

Hash Table

Definition: [Hash Table]

The hash table is a data structure that can be used to store dictionaries.

The hash table consists of a

- an array of n **memory cells**
- and a **hash function** that maps possible “keys” to the location of these cells

n memory cells:

0	**1**	**2**	**...**				**n-1**
		5			1	14	

Key	**Value**
Barcelona	1
Hertha Berlin	5
Real Madrid	14

$h(\text{"Barcelona"})$

$h(\text{"Hertha Berlin"})$

$h(\text{"Real Madrid"})$

The Dictionary Problem

Definition: [The Dictionary Data Structure Problem]

We are given an ordered set of possible keys M.

For example, $M = \{0, 1, \ldots, m-1\}$ (m is very big)

We are also given an array of n memory cells: $C[0], C[1], \ldots, C[n-1]$

We want to introduce the following operations in our data structure:

- **Insert(Key, Value)**
- **Delete(Key)**
- **FindValue(Key)**

The Hash Function

Let $N = \{0, 1, \ldots, n-1\}$, and $M = \{0, 1, \ldots, m-1\}$

If we have access to a $h : M \to N$ function (called hash function), then these operations are simple:

- $\star$ Insert(Key, Value): $C[h(key)] = value$
- $\star$ Delete(Key): $C[h(key)] = none$
- $\star$ FindValue(Key): $C[h(key)]$

Definition: [hash collision]

We say that $h : M \to N$ has a collision in i and j if $h(i) = h(j)$.

In the dictionary data structure we want to avoid collsions, otherwise a memory cell might need to contain multiple values.

If m is big or n is small, collisions will happen...

The Dictionary Problem

Definition: [Perfect hash function]

We say that $h : M \to N$ is a perfect hash function for a set of keys $S \subset M$ if $h(i) \neq h(j)$ for all $i \neq j \in S$.

It means that h doesn't cause any collision among the keys of set S.

The Hash Function

$\star$ A **fixed** h hash function might not work because of collisions.

$\star$ In many applications we want h to be **random** in the sense that for any collection of keys $x_1, x_2, \ldots, x_k$, we want

$$h(x_1), h(x_2), \ldots, h(x_k)$$

to be random, independent, and uniformly distributed on $N = \{0, 1, \ldots, n-1\}$

$\star$ In practice, storing completely random functions $h : M \to N$ is too expensive when m is large.

$\star$ Instead, we will do a trick.

The Hash Function

The trick is that we will work with a family of **fixed, non-random, easy to store and compute** hash functions:

$$\mathcal{H} = \{h : M \to N\}$$

In order to simulate random functions, we will choose h uniformly randomly from $\mathcal{H}$ and study the distributions:

$$\mathbb{P}_{h \in \mathcal{H}}[h(x_1), h(x_2), \ldots, h(x_k)]$$

We want this to look like a joint distribution of k **independent** $U[\{0, 1, \ldots, n-1\}]$ **uniformly** distributed random variables.

To keep the storage cost small, we want $|\mathcal{H}|$ to be small.

Pairwise Independence

Asking for the complete independence of $\mathbb{P}_{h\in\mathcal{H}}[h(x_1), h(x_2), \ldots, h(x_k)]$ can be too difficult, instead we will only require pairwise independence:

$$\mathbb{P}_{h\in\mathcal{H}}[h(x_i) = y_k, h(x_j) = y_l] = \mathbb{P}_{h\in\mathcal{H}}[h(x_i) = y_k]\mathbb{P}_{h\in\mathcal{H}}[h(x_j) = y_l]$$

Homework: Create a distribution of k random variables that are pairwise independent, but jointly not independent.

Strongly 2-Universal Family

Definition: [Strongly 2-Universal Family]

Let $M = \{0, 1, \ldots, m-1\}$

Let $N \doteq \{0, 1, \ldots, n-1\}$

Let $m \geq n$

We say that $\mathcal{H} = \{h : M \to N\}$ is **strongly universal** family

if $\forall x_i \neq x_j,\ y_k, y_l \in N$ we have that

$$\mathbb{P}_{h \in \mathcal{H}}[h(x_i) = y_k, h(x_j) = y_l] = \frac{1}{n^2}$$

Motivation:

If h was a random function, that is

$h(x_i), h(x_j) \sim U[\{0, 1, \ldots, n-1\}]$, and they are independent, then

$$\mathbb{P}_{h \in \mathcal{H}}[h(x_i) = y_k, h(x_j) = y_l] = \mathbb{P}_{h \in \mathcal{H}}[h(x_i) = y_k]\mathbb{P}_{h \in \mathcal{H}}[h(x_j) = y_l]$$
$$= \frac{1}{n}\frac{1}{n} = \frac{1}{n^2}$$

Weakly 2-Universal Family

Often we will require an even less strong property:

Definition: [Weakly 2-Universal Family]

Let $M = \{0, 1, \ldots, m-1\}$, $N = \{0, 1, \ldots, n-1\}$

Let $m \geq n$

We say that $\mathcal{H} = \{h : M \to N\}$ is **weakly universal** family

if $\forall x_i \neq x_j$,we have that

$$\mathbb{P}_{h \in \mathcal{H}}[h(x_i) = h(x_j)] \leq \frac{1}{n}$$

Motivation:

If h was a random function, that is

$h(x_i), h(x_j) \sim U[\{0, 1, \ldots, n-1\}]$, and they are independent, then

$$\mathbb{P}_{h \in \mathcal{H}}[h(x_i) = h(x_j)] = \frac{1}{n}$$

Examples

Lemma:

If $\mathcal{H} = \{h : M \to N\}$ contains all the n^m possible functions, then $\mathcal{H}$ is a strongly 2-universal family.

Example 1 Let $M = \{0, 1\}$ $N \doteq \{0, 1, 2\}$

We have $n^m = 3^2 = 9$ possible functions:

	0	1
h_1	0	0
h_2	0	1
h_3	0	2
h_4	1	0
h_5	1	1
h_6	1	2
h_7	2	0
h_8	2	1
h_9	2	2

Now we have,

$$\mathbb{P}_{h \in \mathcal{H}}[h(x_i) = y_k, h(x_j) = y_l] = \frac{1}{n^2}$$

For example,

$$\mathbb{P}_{h \in \mathcal{H}}[h(0) = 2, h(1) = 1] = \frac{1}{9} = \frac{1}{3^2}$$

Examples

Example 2 Let $M = \{0, 1, 2\}$ $N \doteq \{0, 1\}$

We have $n^m = 2^3 = 8$ possible functions:

	0	**1**	**2**
h_1	0	0	0
h_2	0	0	1
h_3	0	1	0
h_4	0	1	1
h_5	1	0	0
h_6	1	0	1
h_7	1	1	0
h_8	1	1	1

Now we have,

$$\mathbb{P}_{h \in \mathcal{H}}[h(x_i) = y_k, h(x_j) = y_l] = \frac{1}{n^2}$$

For example,

$$\mathbb{P}_{h \in \mathcal{H}}[h(1) = 0, h(2) = 1] = \frac{2}{8} = \frac{1}{4} = \frac{1}{2^2}$$

Examples

Sometimes we don't need to use all the n^m possible functions.

Example 3 Let $M = \{0,1,2\}$ $N \doteq \{0,1,2\}$
Now 9 functions are enough (instead of 27).

	0	**1**	**2**
h_1	0	0	0
h_2	1	1	1
h_3	2	2	2
h_4	0	1	2
h_5	1	2	0
h_6	2	0	1
h_7	0	2	1
h_8	1	0	2
h_9	2	1	0

We have,

$$\mathbb{P}_{h\in\mathcal{H}}[h(x_i) = y_k, h(x_j) = y_l] = \frac{1}{n^2}$$

For example,

$$\mathbb{P}_{h\in\mathcal{H}}[h(0) = 2, h(1) = 1] = \frac{1}{9} = \frac{1}{3^2}$$

$$\mathbb{P}_{h\in\mathcal{H}}[h(0) = 2, h(1) = 2] = \frac{1}{9} = \frac{1}{3^2}$$

How small can family H be?

If we use all the n^m funtions, then $\mathcal{H}$ is too big and requires $\log(n^m) = m \log n$ bits to store the index of the random function picked.

Interestingly, we will see that $O(m^2)$ functions are enough to construct a strongly 2-universal family.

The trick is that we know that according to the Bertrand's postulate there is a prime number p between m and $2m$.
We will use this prime number p.

Constructing Universal Hash Families

Let $N = \{0, 1, \ldots, n-1\}$, $M = \{0, 1, \ldots, m-1\}$.

Let $p \geq m \geq n$ be a prime number.

We will work over the field of $\mathbb{Z}_p = \{0, 1, \ldots, p-1\}$

Let $h_{a,b} : Z_p \rightarrow N$ be defined as:

$$h_{a,b}(x) = ((ax + b \mod p) \mod n) \quad \forall x \in \mathbb{Z}_p$$

Theorem 1 [weakly 2-universal family]

$\mathcal{H} \doteq \{h_{a,b} : M \rightarrow N | 1 \leq a \leq p-1, 0 \leq b \leq p-1\}$ is a weakly 2-universal family.

Note: $|\mathcal{H}| = p(p-1)$

Examples

Example 4 Let $M = \{0,1,2\}$ $N \doteq \{0,1\}$ Let $p = 3$

$p(p-1) = 6$ functions are enough to constract a weakly 2-universal family.

a	b	0	1	2
1	0	0	1	2
1	1	1	2	3
1	2	2	3	4
2	0	0	2	4
2	1	1	3	5
2	2	2	4	6

mod p →

a	b	0	1	2
1	0	0	1	2
1	1	1	2	0
1	2	2	0	1
2	0	0	2	1
2	1	1	0	2
2	2	2	1	0

mod n →

a	b	0	1	2
1	0	0	1	0
1	1	1	0	0
1	2	0	0	1
2	0	0	0	1
2	1	1	0	0
2	2	0	1	0

We can verify that $\mathbb{P}_{h\in\mathcal{H}}[h(x_i) = h(x_j)] \leq \frac{1}{n}$

For example, $\mathbb{P}_{h\in\mathcal{H}}[h(0) = h(1)] = \frac{2}{6} = \frac{1}{3} \leq \frac{1}{2}$

Constructing Universal Hash Families

Theorem 2 [strongly 2-universal family]

Let p be a prime number.

Let $N \doteq \{0, 1, \ldots, p-1\}$ (we have to have $n = p - 1$!)

Let $M \doteq \{0, 1, \ldots, p-1\}$

Let $h_{a,b} : Z_p \to N$ be defined as:

$$h_{a,b}(x) = ax + b \mod p \quad \forall x \in \mathbb{Z}_p$$

Now we have that

$\mathcal{H} \doteq \{h_{a,b} | 0 \leq a \leq p-1, 0 \leq b \leq p-1\}$ is a strongly 2-universal family.

Note: $|\mathcal{H}| = p^2$

See Example 3 for an example

Thanks for your Attention! ☺