Project

Cao Wei
2013311397

Besides the huge quantity of data, it is also a serious problem to deal with the rapid and high dimensional data. In such a background, the data streaming model was proposed. It is a widely used model not only on IT but also in astronomic, communicating and economic fields. The first appearance of this model could be traced back to 20th century, but it is one of the hottest topic in recent years since the scale of the data becomes the bottleneck of modern IT development. Based on such model, the data streaming algorithms have the characteristics of high timeliness, low complexity and approximation.

The approximation is the core of the data streaming algorithm. The real data in internet is actually occupied by the large quantity of redundance. In this section we introduce several classical data streaming algorithms to give a sketch how this model fit the modern data processing.

§1 Conception

In data streaming model, the source signal could be regards as a high dimension vector $\vec{A}$. Initially, $\vec{A}$ equals $\vec{0}$. The data is a continuous tuple sequence $< a_1, ..., a_n >$ denoting a updating of the source signal. According to the type of tuple, it can be classified by:

1. Time Series Model : a_i is an integer indicating update A_i by $a_i = A_i$.
2. Cashier Register Model : a_i is a two-wise tuple $(j, D_i), D_i \geq 0$, which indicating an updating $A_i[j]$ by $A_{i-1}[j] + D_i$
3. Turnstile Model : a_i is a two-wise tuple $(j, D_i), D_i \in R$, which indicating an updating $A_i[j]$ by $A_{i-1}[j] + D_i$

This paper only concern about the cashier model. Consider an IP tracker, with each visiting counted once by $(IP, 1)$, thus it is a representational application of cashier model. Moreover, assume all the data is pass-efficient.

§2 Methods

The typical methods in streaming model consist of *Sampling* and *Sketching*. *Sampling* method will draw the data with a specific probability and the distribution of data will be reconstruct by the samples. While the *Sketching* method mapping the arbitrary data with a hash function to a length fixed space in order to decrease the space complexity. Moreover the sketching method is simple to build but well performed. Especially, the error of approximation could be bounded with a small factor of L_p norm.

Human beings could be regarded as a "biological" sketching machine. The sensors of human receive a consequently high dimensional signal(such as audio and video signal) from the nature rapidly and process it with sketching method. Therefore people usually could not remember every thing happened before, but they might have a vague idea about that. Differ from sampling, the sketching method is sensitive to abnormal data which implies the sketching could be used into anomaly detection.

§2.1 Flajolet-Martin

Flajolet-Martin is used to estimate distinct elements of data. Suppose the sequence of streaming equals $\delta =< a_1, ..., a_n >$, for each a_i we have $a_i = (j \in [n], D_i = 1)$. Therefore it is a streaming description of frequency vector $f = (f_1, .., f_n)$ and the result algorithm's output should be $d = |\{j : f_j > 0\}|$.

Since the dimension of source signal could be unsolvable large, then the algorithm is supposed to output an approximation with small error bounded. Formally, suppose the output is $\hat{d}$, for any $\epsilon, \delta > 0$, $Pr[|\hat{d} - d| > \epsilon]$ should be less equal than δ.

Let $zeor(x)$ be the quantity of 0s in the tail of binary representation. Here we provide the algorithm:

Flajolet-Martin Algorithm

1. Initialization

   Choose a hash function $h : [n] \to [n]$ from 2-wise independent hash function family.

   Initialize $z = 0$

2. Processing Data

   Update $z = \max\{z, zeros(h(j))\}$

3. Output

   $2^{z+\frac{1}{2}}$

Although Flajolet-Martin Algorithm is a simple algorithm but the reason it works is not trivial. To analyze the algorithm, suppose $z = t$ when the algorithm terminate. Let $Y_r = \sum_{j:f_i>0} X_{r,j}$, then $Y_r \geq 0 \Leftrightarrow t \geq r$. Equivalently, $Y_r = 0 \Leftrightarrow t \leq r-1$. Since h maps an arbitrary number in $[n]$ to a binary string with length $log(n)$. Therefore:

$$E[X_{r,j}] = Pr[zeros(h(j)) \geq r] = Pr[2^r divides h(j)] = \frac{1}{2^r}$$

$$E[Y_r] = \sum_{j:f_i>0} E[X_{r,j}] = \frac{d}{2^r}$$

Also notice h is pair-wise independent, we have

$$Var[Y_r] = \sum_{j:f_j>0} Var[X_{r,j}] \leq \sum_{j:f_j>0} E[X_{r,j}^2] = \sum_{j:f_j>0} E[X_{r,j}] = \frac{d}{2^r}$$

Substitute the equation to *Markov Inequality* and *Chebyshev Inequality*, it implies:

$$Pr[Y_r > 0] = Pr[Y_r \geq 1] \leq \frac{E[Y_r]}{1} = \frac{d}{2^r}$$

and

$$Pr[Y_r = 0] = Pr[|Y_r - E[Y_r]| \geq \frac{d}{2^r}] \leq \frac{2^r}{d}$$

To give a reasonable error bound of $\hat{d} = 2^{t+\frac{1}{2}}$, let a be the smallest integer satisfy $2^{t+\frac{1}{2}} \geq 3d$, b be the largest integer satisfy $2^{t+\frac{1}{2}} \leq \frac{d}{3}$. Thus it is guaranteed with probability at least 53% $\hat{d}$ is in interval $[\frac{d}{3}, 3d]$.

Since the algorithm is supposed to bound the error with arbitrary ϵ. A statistical trick called *Median Trick* will be used here. The intuition of Median Trick is to maintain k procedures parallelly and promise the distribution of k estimator is approximate Gaussian distribution. Therefore we could pick $\hat{d}_{k/2}$ as the final estimator instead. More precisely, we need introduce a crucial bound in probability theory, the *Chernoff Bound*.

Chernoff Bound: $X_1, X_2, ..., X_n$ are random variables follows distribution with parameters p. Even S denote more than half of X equals 1. Then $Pr(S) \geq 1 - exp\{-\frac{1}{2p}n(p-\frac{1}{2})^2\}$.

Let X_i denote $\hat{d}_i \geq 3d$, substitute $n = k, p \leq \frac{\sqrt{2}}{3}$ we have $Pr(S) = 2^{-\Omega(k)} = \delta$. Maintain $k = \Theta(log(\frac{1}{\delta}))$ procedures parallelly, and output $\hat{d}_{1/2}$, the algorithm gives a $(O(1), \delta)$ approximation with space complexity $O(log(\frac{1}{\delta})log(n))$.

§2.2 Count Sketch

Count Sketch is a crucial streaming data structure. It supports point, range, inner product and even quantile queries.

Restricted in cashier model, the algorithm maintain a t-rows k-columns table. Each row contains a vector in k dimension indicating the projection space of the hash function. Utilize the median trick, the algorithm is provided:

Count Sketch Algorithm For Point Query

1. Initialization

   build a table C with t rows and k columns, $t = \lceil log(\frac{1}{\delta}) \rceil$, $k = \frac{3}{\epsilon^2}$. Set all the elements as zero.

   Select t independent hash functions $h : [n] \to [k]$ from 2-universal randomly

   Select t independent hash functions $g : [n] \to \{-1, +1\}$ from 2-universal randomly

2. Processing Data

   $$for\ i = 1 \to\ t\ \textbf{do}\ C[i][h_i(j)]\ \leftarrow\ C[i][h_i(j)] + cg_i(j)$$

3. Output

   $$\hat{f_a} = median_{1 \le i \le t}\ g_i(a) C[i][h_i(a)]$$

To analyze the algorithm, consider a specific row of the table C_t. Let $X_t = \hat{f_a}$ be the output of input a, the indicator variable Y_j indicate $h(j) = h(a)$. Since $C_t[h(a)]$ is updated only when $h_t(j) = h_t(a)$, the accumulated contribution is $f_j \cdot g_t(j)$, Therefore

$$X = g_t(a) \sum_{j=1}^{n} f_j g_t(j) Y_j = f_a + \sum_{j \in [n] \setminus \{a\}} f_j g_t(a) Y_j$$

According to the properties of expectation, we have

$$E[g_t(j) Y_j] = E[g_t(j)] E[Y_j] = 0$$

which implies

$$E[X_t] = f_a + \sum_{j \in [n] \setminus \{a\}} f_j g_t(a) Y_j = f_a + 0 = f_a$$

As it shows the output is an unbiased estimator of f_a. Since $g(a)^2 = 1$, it implies $Var[x] = \sum_{j \in [n] \setminus \{a\}} \frac{f_j^2}{k} = \frac{\|\mathbf{f}\|_2^2 - f_a^2}{k}$. Take it into Chernoff bound and let $t = O(log(\frac{1}{\delta}))$, then for any given $\epsilon, \delta > 0$, count sketch promise $Pr[|\hat{f_a} - f_a| \ge \epsilon \parallel \mathbf{f}_{-a} \parallel_2] \le \delta$.