CS369G: Algorithmic Techniques for Big Data Spring 2015-2016

Lecture 15: Lower bounds for streaming algorithms

Prof. Moses Charikar *Scribes: Hongyang Zhang*

1 Overview

Last time we studied two problems:

- *Disjointness:* Alice and Bob each has a subset of $[n]$. They want to check if the two sets are disjoint or not.
- *INDEX:* Alice has a subset of elements in $[n]$ and Bob holds an element (index) $i \in [n]$. How much information does Alice need to send to Bob, so that Bob can find out whether Alice's set includes the i-th element?

For both problems, we showed that the one-way communication complexity is $\Omega(n)$, even for randomized protocols. In this lecture, we will prove that in order to obtain a $(1+\varepsilon)-$approxation of F_0, you need at least $\Omega(\frac{1}{\varepsilon^2})$ space. This will be done using a reduction from Gap-Hamming.

2 The Gap-Hamming problem

We will first prove a special case when $\varepsilon = \frac{1}{\sqrt{n}}$. Consider reducing the problem of approximating F_0 to Disjointness. Suppose that there are two sets $A, B \subset [n]$ such that they intersect on at most one element (INDEX), if we only have a $(1+\frac{1}{\sqrt{n}})-$approximation of F_0, then it wouldn't be able to tell whether A and B are disjoint or not. Thus, we need to add some slack between the two cases, to be able to use approximation of F_0 to construct a communication protocol.

Gap-Hamming. Given two inputs $x \in \{0,1\}^n$ and $y \in \{0,1\}^n$ we define the function

$$\text{GapHam}_c(x,y) = \begin{cases} 1 & \text{if } d_H(x,y) \le \frac{n}{2} - c\sqrt{n} \\ \text{undefined} & \text{otherwise} \\ 0 & \text{if } d_H(x,y) \ge \frac{n}{2} + c\sqrt{n} \end{cases}$$

Claim 1. *If Gap-Hamming needs $\Omega(n)$ bits of communication, then approximating F_0 within $1+\frac{1}{\sqrt{n}}$ needs $\Omega(n)$ space.*

Proof. If there is a protocol such that Alice sends $o(n)$ bits communication to Bob, and Bob can figure out a $(1+\frac{1}{\sqrt{n}})-$approximation of F_0, then we claim that there is a $o(n)$ communication

protocol to solve the Gap-Hamming problem. Let S_x denote the set of nonzero indices of x and S_y denote the set of nonzero indices of y. Let $T = |S_x \cup S_y|$ denote the size of their union. Then,

$$\begin{aligned} T &= |S_x| + |S_y| - |S_x \cap S_y| \\ d_H(x,y) &= n - |S_x \cap S_y| \end{aligned}$$

Combining the two equations we get that:

$$T = d_H(x,y) + |S_x| + |S_y| - n$$

If Alice and Bob can get $\tilde{T}$ such that $|\tilde{T} - T| \leq T/\sqrt{n}$, then

- if $d_H(x,y) \geq (\frac{1}{2} + \frac{2}{\sqrt{n}}) \cdot n$, we get $T \geq |S_x| + |S_y| - \frac{n}{2} + 2\sqrt{n}$, hence

$$|S_x| + |S_y| \leq \left(1 - \frac{1}{\sqrt{n}}\right)^{-1} \tilde{T} + \frac{n}{2} - 2\sqrt{n} := L_n(\tilde{T})$$

- if $d_H(x,y) \leq (\frac{1}{2} - \frac{2}{\sqrt{n}}) \cdot n$, we get $T \leq |S_x| + |S_y| - \frac{n}{2} - 2\sqrt{n}$, hence

$$|S_x| + |S_y| \geq \left(1 + \frac{1}{\sqrt{n}}\right)^{-1} \tilde{T} + \frac{n}{2} + 2\sqrt{n} := U_n(\tilde{T})$$

Since, $\tilde{T} \leq n$ (there can be at most n different elements) we have that for all $n \geq 3$:

$$U_n(\tilde{T}) - L_n(\tilde{T}) \geq \left(4 - 2\frac{n}{n-1}\right)\sqrt{n} > 0$$

Therefore, the two cases can be distinguished if Alice sends also $|S_x|$ to Bob using extra $\log n$ bits, as Bob can use the $\tilde{T}, |S_x|, |S_y|$ to see whether $|S_x| + |S_y| \leq L_n(\tilde{T})$ or $|S_x| + |S_y| \geq U_n(\tilde{T})$. □

3 Reduction from INDEX

Now we prove that solving the Gap-Hamming problem requires $\Omega(n)$ bits of communication. We will reduce the Gap-Hamming problem to INDEX.

INDEX. Given a string $x \in \{0,1\}^n$ and an index $i \in [n]$ compute $INDEX(x,i) = x_i \in \{0,1\}$.

We will assume that n is odd and that both Alice and Bob have common access to an infinitely long random string r (a.k.a. the public-coin model, see Tim's lecture notes for more comments versus private-coin models).

3.1 Communication Protocol for Index using Gap-Hamming

Alice and Bob will generate without communication (using the public coins) a valid input $(\tilde{x}, \tilde{y})$ to the Gap-Hamming problem. Then, the lower bound of $\Omega(n)$ on the communication complexity of index will carry on to Gap-Hamming as well.

At a high level, Alice and Bob will respectively generate a sequence of independent "random" bits $(\tilde{x}_j, \tilde{y}_j)$ for $j \in [m]$ that are slightly correlated if $x_i = 1$ and anti-correlated if $x_i = 0$, where i is Bob's index and x is Alice's original string.

The intuition of why this is possible is that, if we fix Alice's vector x and then look at the distance from a random string, if the random string is closer than $n/2$ then the bits of the random string r we just generated happen to be slightly correlated with Alice's string. Therefore, Bob by outputting the bit r_i in this case produces a bit that is slightly biased to agree with x_i. Then by repeating this process independently enough times we make this correlation detectable.

To construct the first random bit $\tilde{x}_1, \tilde{y}_1$ consider the vector z consisting of the first n bits of the random string r.

- *Alice:* If $d_H(x, z) < n/2$, set $\tilde{x}_1 = 1$; else set $\tilde{x}_1 = 0$.
- *Bob:* set $\tilde{y}_1 = z_i$.

We then repeat this process n times and generate strings $\tilde{x}, \tilde{y}$ that are fed as input to any communication protocol for GapHamming for which we report it's output as our result.

3.2 Analysis of the protocol

We consider the correlation between $\tilde{x}_1$ and $\tilde{y}_1$. Let x_{-i} denote the string without the i-thy coordinate.

- *No correlation:* if the Hamming distance between x_{-i} and r_{-i} is strictly less than $(n-1)/2$ or at least $(n+1)/2$, then $\tilde{x}_1 = 1$ independently of z_i. Hence $\Pr(\tilde{x}_1(z) = \tilde{y}_1) = \Pr(\tilde{x}_1 = z_i) = \frac{1}{2}$, this follows because $\tilde{x}_1$ is not a function of z_i.
- *Correlation:* otherwise, the Hamming distance between x_{-i} and z_{-i} is exactly $(n-1)/2$. This happens with probability $\binom{n-1}{(n-1)/2}/2^{n-1} = \Theta(1/\sqrt{(}n))$ using Stirling's approximation of the factorial. In this case,
  (a) *Positive:* if $x_i = 1$, then $\tilde{x}_1 = \tilde{y}_1 = z_i$.
  (b) *Negative:* If $x_i = 0$, then $\tilde{x}_i = 1 - z_i$ and $\tilde{y}_1 = z_i$.

Let A denote the event that $d_H(x_{-i}, z_{-i}) = \frac{n-1}{2}$. Assuming that $\Pr(A_i) \approx d/\sqrt{n}$ and applying total probability law we get:

$$\Pr(\tilde{x}_1 = \tilde{y}_1) = \begin{cases} \frac{1}{2}\left(1 - \frac{d}{\sqrt{n}}\right) + 1 \cdot \frac{d}{\sqrt{n}} & \text{if } x_i = 1 \\ \frac{1}{2}\left(1 - \frac{d}{\sqrt{n}}\right) + 0 \cdot \frac{d}{\sqrt{n}} & \text{if } x_i = 0 \end{cases}$$

If we repeat the above process for $m = cn$ times, then using Chernoff bounds we can show that with high probability we obtain two strings x' and y', such that:

- if $x_i = 1$, then $d_H(x', y') \leq \frac{n}{2} - c/\sqrt{n}$;
- if $x_i = 0$, then $d_H(x', y') \geq \frac{n}{2} + c/\sqrt{n}$.

So if we have a communication protocol for solving Gap-Hamming, then we can first generating x' and y' and then use that protocol to send x' (y').

Remark 1: for more general ε, one can use a padding argument (see Tim's lecture notes for more details).

Remark 2: The following is an interesting question: you have two strings and you want to determine whether the two strings are close (Hamming distance being at most x) or far away (Hamming distance being at least y). Depending on different parameter settings, how do we solve this problem? This is closely related to the Locaility Sensitive Hashing (LSH) that we will talk about later on in the course.

4 Lower bounds for graph connectivity in the streaming model

You are given a graph in the streaming model. Given two vertices s and t, you want to check if s and t are in the same connected component or not. We will show that this needs $\Omega(n)$ bits of space, where n is the number of vertices of the graph.

We reduce Disjointness to Graph connectivity. Given two strings $x, y \in \{0,1\}^n$, construct a graph $G = (V, E)$ as follows:

- Nodes: $V = [n] \cup \{s, t\}$, a node for each coordinate and two distinguished vertices s, t.
- Edges: $E = \{\{s, i\} | \forall i : x_i = 1\} \cup \{\{i, t\} | \forall i : y_i = 1\}$ connect s (resp. t) to all nodes $i \in [n]$ where $x_i = 1$ (resp. $y_i = 1$).

Then s and t are connected in G if and only if x, y are not disjoint.

One can ask the question what else can we do if we can have multiple passes over a graph stream. In general, if we have a limited amount of storage, how many passes do we need in order to answer whether s and t are connected or not? One known result is that if we can have p passes over a graph stream, then we need $\Omega(n/p)$ bits of space to solve connectivity.

5 Lower bounds for higher order frequency moments

The Multi-party set disjointness problem is the following:

1. t players $A_1, \ldots, A_t$
2. each with their private string $x^{(i)} \in \{0,1\}^n, \forall i \in [t]$.
3. for all $i \in [n-1]$, player A_i sends a piece of information to A_{i+1}.

4. In the end, player A_n outputs a result.

Suppose that the input falls in either of the following two cases: In the Yes case, x_i and x_j are disjoint for all i, j; In the No case, there exists $a \in [n]$, such that $x_i \cap x_j = \{a\}$, for all i, j.

$$\text{MultiDISJ}_t(x^{(1)}, \ldots, x^{(t)}) = \begin{cases} 1 & \text{if} \quad \text{DISJ}(x^{(i)}, x^{(j)}) = 1, \forall i \neq j \in [t] \\ 0 & \text{if} \quad \exists a \in [n], x_a^{(i)} = 1, \forall i \in [t] \end{cases}$$

It can be shown that the One-way communication complexity for this problem is $\Omega(\frac{n}{t})$. This can be turned to obtain a $\Omega(n^{1-\frac{2}{k}})$ space lower bound for estimating F_k.