CS 15-859: Algorithms for Big Data Fall 2020

Lecture 11 — 11/19/20

Prof. David Woodruff *Scribe: Jessica Lee*

1 Randomized 1-Way Communication Complexity for Indexing

We consider the problem of 1-way communication complexity of Index, where Alice, who has a string $x \in \{0,1\}^n$, sends a single message M to Bob, and Bob, given M and $j \in [n]$, should output x_j with probability at least $\frac{2}{3}$. Note that this probability is computed over all coin tosses, not inputs, meaning that $\forall x, \forall j, Pr_{\text{coin tosses}}[\text{Bob outputs } x_j] \geq \frac{2}{3}$.

We want to show the lower bound on the number of bits that we have to transfer in order to solve the problem is $\Omega(n)$. We first consider a uniform distribution μ on $X \in \{0,1\}^n$. We know that Bob will output an estimate X'_j, where we know that $Pr[X'_j = X_j] \geq \frac{2}{3}$. However, what can we gather about X_j from M, or how much information is revealed? We can apply Fano's inequality because X_j and X'_j are independent given M by the Markov Chain $X \to M \to X'$.

$$H(X_j|M) \leq H(P_e) + P_e(\log_2(|X|-1)) \qquad \text{[Def'n of Fano's Inequality]} \tag{1}$$

$$\leq H(\frac{1}{3}) + \frac{1}{3}(\log_2(2-1)) \qquad [|X| = 2 \text{ b/c binary, } P_e = \frac{1}{3}] \tag{2}$$

$$\leq H(\frac{1}{3}) < 1 \tag{3}$$

Since the conditional entropy is less than 1, that means the message does intuitively reveal alot of information about X.

We now consider the mutual information $I(M;X)$, remembering that M is a random variable that depends on X, and X has a uniform distribution. We now apply Chain Rule in order to derive a lower bound. We use $X_{<i}$ to denote the bits preceding bit i, and do recall that X is a list of bits.

$$I(X;M) = \sum_i I(X_i; M|X_{<i}) \qquad \text{[list of bits independent]} \tag{4}$$

$$= \sum_i H(X_i|X_{<i}) - H(X_i|M, X_{<i}) \qquad \text{[Def'n of Mutual Information]} \tag{5}$$

$$= n - \sum_i H(X_i|M, X_{<i}) \qquad [X_i \text{ and } X_{<i} \text{ are independent}] \tag{6}$$

$$\geq n - \sum_i H(X_i|M) \qquad [H(X_i|M, X_{<i}) \leq H(X_i|M)] \tag{7}$$

$$\geq n - H(\frac{1}{3})n \qquad \text{[Fano's Inequality (line 3)]} \tag{8}$$

$$= \Omega(n) \tag{9}$$

Thus, we find that the mutual information is at least $\Omega(n)$ bits. How does this relate to the communication complexity, M? Consider $|M| = \ell$ bits, which means there are up to 2^ℓ possibilities. If we consider $H(M)$, we know that M is uniform, so $H(M) \leq log_2(2^\ell) = \ell$. Therefore, we have that $|M| \geq H(M)$. Additionally, we have that $H(M) \geq I(X;M)$ by definition of mutual information, which implies $I(X;M) = H(M) - H(M|X)$. Therefore, by line 9, we have

$$|M| \geq H(M) \geq I(X;Y) = \Omega(n) \tag{10}$$

This formalizes that the length of Alice's message has to be at least $\Omega(n)$ bits.

2 Typical Communication Reduction

It is natural that we want to reduce a problem we are solving to another (such as Index) to attain a lower bound. Here's a natural framework for that.

Participant	Alice	Bob
Data	$a \in \{0,1\}^n$	$b \in \{0,1\}^n$
Stream	$s(a)$	$s(b)$

1. Run streaming algorithm on $s(a)$, and transmit the state of $Alg(s(a))$ to Bob.
2. Bob computes $Alg(s(a), s(b))$
3. If Bob solves $g(a,b)$, the space complexity of Alg is at least the 1-way communication complexity of g

3 Example: Distinct Elements

We want to solve the problem given $a_1, ..., a_m \in [n]$, how many distinct (non-duplicate) elements are there? We want to show that it requires at least $\Omega(n)$ bits of communication complexity as a lower bound.

We will reduce Index problem to Distinct Elements. Recall that in the index problem, Alice has a bit string $x \in \{0,1\}^n$, and Bob has an index $i \in [n]$. Bob wants to know if $x_i = 1$. The reduction is as follows.

Set $s(a) = i_1...i_r$, where i_j appears iff $x_{i_j} = 1$. We set $s(b) = i$. Now, we case on the cases on what x_i will be.

$$x_i = \begin{cases} 0 & \text{if } Alg(s(a), s(b)) = Alg(s(a)) + 1 \\ 1 & \text{otherwise} \end{cases} \tag{11}$$

In the case where $Alg(s(a), s(b)) = Alg(s(a)) + 1$, then that means when we concatenate $s(b)$ into the stream, then we have one more distinct element than originally. Consider if $x_i = 1$, this means i appeared in $s(a)$, then that means when we concatenated $s(b)$, i was no longer a distinct element. Therefore, if $x_i = 0$, we have one more distinct element than before!

This has now proved that the space complexity of Alg, has lat least the 1-way communication complexity of Index.

4 Strengthening Index: Augmented Indexing

We will now beef up the Index problem to provide Bob with more information than before. Bob will have access to all the previous bits up to x_{i-1}. Bob still wants to learn x_i from Alice.

Participant	Alice	Bob
Information	$x \in \{0,1\}^n$	$i \in [n]$ and $x_1 ... x_{i-1}$
Stream	$s(a)$	$s(b)$

We will still show that this problem has a lower bound of $\Omega(n)$. We will provide a similar proof to Index.

$$
\begin{aligned}
I(X; M) &= \sum_i I(X_i; M | X_{<i}) && \text{[list of bits independent]} && (12)\\
&= \sum_i H(X_i | X_{<i}) - H(X_i | M, X_{<i}) && \text{[Def'n of Mutual Information]} && (13)\\
&= n - \sum_i H(X_i | M, X_{<i}) && [X_i \text{ and } X_{<i} \text{ are independent}] && (14)\\
& && && (15)
\end{aligned}
$$

This time however, we will use the Markov Chain $X \to X_{<i}, M \to X_i'$. using Fano's Inequality, we attain $H(X_i|M) \leq H(\delta) + \delta \log_2(2-1) = H(\delta)$. We continue our lower bound proof below:

$$
\begin{aligned}
I(X; M) &\geq n - H(\delta)n && (16)\\
&\geq n(1 - H(\delta)) && (17)
\end{aligned}
$$

Therefore, we can lower bound the communication complexity (CC) generally by:

$$
\begin{aligned}
CC_\delta(\text{Augmented\_Index}) \geq I(M; X) \geq n(1 - H(\delta)) && (18)\\
&& (19)
\end{aligned}
$$

5 $\log(n)$ Bit Lower Bound for Estimating Norms

We will now consider the problem where Alice has $\log(n)$ bits, as an input into Augmented Index. We want to set a lower bound to estimating a norm. A basic question in streaming is to maintain a counter, which is a stream of updates $c \leftarrow c+1, c \leftarrow c-1, c \leftarrow c-1$. We want to ask ourselves, can we approximate the counter to a factor of 2. To do so exactly, we need at least $\log(n)$ bits. If we were to output a 2-approximation, to write down the answer we only need $log(log(n))$ bits; however, our following proof will show that the problem is harder than just the space needed to write down the answer, meaning it is non-trivial to solve the approximation case as well.

Alice will create a vector v, with a single coordinate equal to $\sum_i^{\log(n)} 10^j x_j$. This means approximating the norm of a vector $v = (c, 0, 0, ..., 0)$, will imply that any p-norm $||v||_2, ||v||_1, ...||v||_p$ will be equivalent to estimating c. Alice will send bob the state of the data stream algorithm after feeding in v.

Bob has $i \in [\log(n)]$, and $x_{i+1}, x_{i+2}, ..., x_{\log(n)}$. Bob now creates the vector $w = \sum_{j>i} 10^j$. Now, $Alg(v-w) = Alg(\sum_{j \leq i} 10^j x_j)$.

$$x_i = \begin{cases} 1 & \text{if } Alg(v-w) \geq \frac{10^i}{2} \\ 0 & \text{otherwise} \end{cases} \tag{20}$$

So, if $x_i = 0$, So, we have that $Alg(v-w) \leq 1 + 10 + 100 + ... + 10^{i-1} < \frac{2}{9}10^i$. If $x_i = 1$, so we have $Alg(v-w) \geq \frac{10^i}{2}$. There is a gap between the bounds even with the 2-approximation, therefore, our algorithm holds.

This shows that there is a $log(n)$ bit lower bound for estimating norms.

6 $\frac{1}{\varepsilon^2}$ Bit Lower Bound for Estimating Norms

We now want to reduce Index to the Gap Hamming Problem in order to prove a $\frac{1}{\varepsilon^2}$ lower bound for estimating norms., and we now follow the proof proposed in [1].

Definition. Hamming Distance $\Delta(x, y)$: The number of coordinates i such that $x_i \neq y_i$.

Definition. Gap Hamming Problem: Given two arrays of length n, we have the promise that one of the two occurs: $\Delta(x, y) > \frac{n}{2} + 2\varepsilon n$ or $\Delta(x, y) < \frac{n}{2} + \varepsilon n$, with the goal to figure out which of the two cases occur.

Definition. Public Coin: A sequence of vectors $r^1, ..., r^t$, each in $\{0, 1\}^t$. We can also represent this as a $t \times t$ matrix of uniformly random bits, where each row i corresponds to r^i.

Consider Alice has a stream x of length t, where $t = O(\frac{1}{\varepsilon^2})$, so $a = \{0, 1\}^t$. Then Bob has index i, where $b = \{0, 1\}^t$. We construct a, b from the public coin as follows:

$$a_k = \text{Majority}_{\text{j such that } x_j = 1}\ r_j^k \tag{21}$$

$$b_k = r_i^k \tag{22}$$

In plain terms, for b, we just take the i-th column of the public coin. For a, we take a subset of the columns, only those such that $x_j = 1$, and we take the majority of the bits in each row.

Now let's consider when $x_i = 0$. Noting that the construction of a and b are independent since the i-th column is not considered in x_i, we know that $\mathbb{E}[\Delta(a,b)] = \frac{t}{2}$, since each element has a probability of $\frac{1}{2}$ of matching.

Now if $x_i = 1$, then a, b are dependent. consider the probability of each bit being a majority bit. The weight of a vector are the number of elements that are 1, and $k = weight(x)$.

$$\begin{aligned}
\mathbf{Pr}[\text{majority}(Z_1, ..., Z_k) = Z_1]] &\approx \frac{1}{2} + \frac{\binom{k-1}{\frac{k-1}{2}}}{2^k} && (23)\\
&\approx \frac{1}{2} + \Theta(\frac{1}{\sqrt{k}}) && (24)\\
&\approx \frac{1}{2} + \Theta(\frac{1}{\sqrt{t}}) \qquad [\text{weight at most } t] && (25)\\
&\approx \frac{1}{2} + \Theta(\epsilon) && (26)\\
& && (27)
\end{aligned}$$

Therefore, if we compute expected Hamming Distance, we get

$$\begin{aligned}
\mathbb{E}[\Delta(a,b)] &= \frac{t}{2} - \Theta(\epsilon t) && (28)\\
&= \frac{t}{2} - \Theta(\frac{1}{\varepsilon}) && (29)
\end{aligned}$$

We can now write the expectation in a more general form without casing on the value of x_i.

$$\mathbb{E}[\Delta(a,b)] = \frac{t}{2} - x_i\sqrt{t} \tag{30}$$

This verifies that given the Gap Hamming Algorithm, we can solve Indexing. Since we set $t = \Theta(\frac{1}{\varepsilon^2})$, we find that the lower bound for gap hamming is $\frac{1}{\varepsilon^2}$ by this derivation.

6.1 But we used a public coin...?

Is Indexing still hard even with randomness? In this section we want to show that even if we possess a public coin, the lower bound for indexing remains the same. Assume standard indexing parameters.

$$
\begin{aligned}
I(X; M|R) &= \sum_i I(X_i; M|X_{<i}, R) && \text{[list of bits independent]} && (31)\\
&= \sum_i H(X_i|X_{<i}, R) - H(X_i|M, X_{<i}, R) && \text{[Def'n of Mutual Information]} && (32)\\
&= H(X_i|R) - \sum_i H(X_i|M, R) && [X_i \text{ and } X_{<i} \text{ are independent}] && (33)\\
&\geq n - \sum_i H(X_i|M, R) && [H(X_i|M, X_{<i}) \leq H(X_i|M)] && (34)\\
&\geq n - H(\delta)n && \text{[Fano's Inequality]} && (35)\\
&= \Omega(n) && && (36)
\end{aligned}
$$

where line 35 follows from Fano's Inequality here: $H(X_i|M, R) \leq H(\delta)$.

So, we find that $CC_\delta(Index) \geq I(X; M|R) \geq n(1 - H(\delta))$.

References

[1] Thathachar S Jayram, Ravi Kumar, and D Sivakumar. The one-way communication complexity of hamming distance. *Theory of Computing*, 4(1):129–135, 2008.