DIMACS Technical Report 2005-33

Estimating Entropy and Entropy Norm on Data Streams

by

Amit Chakrabarti[1] Khanh Do Ba[2] S. Muthukrishnan[3]

[1]Supported by an NSF CAREER award and Dartmouth College startup funds.
[2]Work partly done while visiting DIMACS in the REU program, supported by NSF ITR 0220280, DMS 0354600, and a Dean of Faculty Fellowship from Dartmouth College.
[3]Supported by NSF ITR 0220280 and DMS 0354600.

ABSTRACT

We consider the problem of computing information theoretic functions such as entropy on a data stream, using sublinear space.

Our first result deals with a measure we call the "entropy norm" of an input stream: it is closely related to entropy but is structurally similar to the well-studied notion of frequency moments. We give a polylogarithmic space one-pass algorithm for estimating this norm under certain conditions on the input stream. We also prove a lower bound that rules out such an algorithm if these conditions do not hold.

Our second result is a sublinear space one-pass algorithm for estimating the empirical entropy of an input stream. For a stream of m items and a given real parameter α, our algorithm uses space $\widetilde{O}(m^{2\alpha})$ and provides an approximation of $1/\alpha$ in the worst case and $(1+\varepsilon)$ in "most" cases. All our algorithms are quite simple.

1 Introduction

Algorithms for computational problems on data streams have been the focus of plenty of recent research in several communities, such as theory, databases and networks [1, 5, 2, 7]. In this model of computation, the input is a stream of "items" that is too long to be stored completely in memory, and a typical problem involves computing some statistics on this stream. The main challenge is to design algorithms that are efficient not only in terms of running time, but also in terms of space (i.e., memory usage): sublinear space is a must and polylogarithmic space is often the goal.

The seminal paper of Alon, Matias and Szegedy [1] considered the problem of estimating the *frequency moments* of the input stream: if a stream contains m_i occurrences of item i (for $1 \leq i \leq n$), its k^{th} frequency moment is denoted F_k and is defined by $F_k := \sum_{i=1}^{n} m_i^k$. Alon et al. showed that F_k could be estimated arbitrarily well in sublinear space for all nonnegative integers k and in polylogarithmic (in m and n) space for $k \in \{0, 1, 2\}$. Their algorithmic results were subsequently improved by Coppersmith and Kumar [3] and Indyk and Woodruff [6].

In this work, we first consider a somewhat related statistic of the input stream, inspired by the classic information theoretic notion of entropy. We consider the *entropy norm* of the stream, denoted F_H and defined by $F_H := \sum_{i=1}^{n} m_i \lg m_i$.[1] We prove (see Theorem 2.2) that F_H can be estimated arbitrarily well in polylogarithmic space provided its value is not "too small," a condition that is satisfied if, e.g., the input stream is at least twice as long as the number of distinct items in it. We also prove (see Theorem 2.4) that F_H cannot be estimated well in polylogarithmic space if its value *is* "too small."

Second, we consider the estimation of entropy itself, as opposed to the entropy norm. Any input stream implicitly defines an *empirical* probability distribution on the set of items it contains; the probability of item i being m_i/m, where m is the length of the stream. The *empirical entropy* of the stream, denoted H, is defined to be the entropy of this probability distribution:

$$H \;:=\; \sum_{i=1}^{n} (m_i/m) \lg(m/m_i) \;=\; \lg m - F_H/m\,. \tag{1}$$

An algorithm that computes F_H exactly clearly suffices to compute H as well. However, since we are only able to approximate F_H in the data stream model, we need new techniques to estimate H. We prove (see Theorem 3.1) that H can be approximated using sublinear space. Although the space usage is not polylogarithmic in general, our algorithm provides a tradeoff between space and approximation factor and can be tuned to use space arbitrarily close to polylogarithmic.

Both entropy and entropy norm are natural statistics to approximate on data streams. In addition, they are used in profiling IP traffic on communication networks and for anomaly detection [8]. Our algorithms are quite simple and may prove useful in real IP network traffic analysis systems.

[1] Throughout this paper "lg" denotes logarithm to the base 2.

2 Estimating the Entropy Norm

In this section we present a polylogarithmic (in m and n) space $(1+\varepsilon)$-approximation algorithm for entropy norm that assumes the norm is sufficiently large, and prove a matching lower bound if the norm is in fact not as large.

2.1 Upper Bound

Our algorithm is inspired by the work of Alon et al. [1]. Their first algorithm, for the frequency moments F_k, has the following nice structure to it (some of the terminology is ours). A subroutine computes a *basic estimator*, which is a random variable X whose mean is exactly the quantity we seek and whose variance is small. The algorithm itself uses this subroutine to maintain $s_1 s_2$ indepedent basic estimators $\{X_{ij} : 1 \le i \le s_1, 1 \le j \le s_2\}$, where each X_{ij} is distributed identically to X. It then outputs a *final estimator* Y defined by

$$Y \;:=\; \operatorname*{median}_{1 \le j \le s_2} \left(\frac{1}{s_1} \sum_{i=1}^{s_1} X_{ij} \right)$$

The following lemma, implicit in [1], gives a guarantee on the quality of this final estimator.

Lemma 2.1. *Let $\mu := \mathrm{E}[X]$. If $s_1 \ge 8\,\mathrm{Var}[X]/(\varepsilon^2\mu^2)$ and $s_2 = 4\lg(1/\delta)$, then for any $\varepsilon, \delta \in (0, 1]$, the above final estimator deviates from μ by no more than $\varepsilon\mu$ with probability at least $1 - \delta$. The above algorithm can be implemented to use space $O(S \log(1/\delta)\,\mathrm{Var}[X]/(\varepsilon^2\mu^2))$, provided the basic estimator can be computed using space at most S.*

Proof. The claim about the space usage is immediate from the structure of the algorithm. Let $Y_j = \frac{1}{s_1}\sum_{i=1}^{s_1} X_{ij}$. Then $\mathrm{E}[Y_j] = \mu$ and $\mathrm{Var}[Y_j] = \mathrm{Var}[X]/s_1 \le \varepsilon^2\mu^2/8$. Applying Chebyshev's Inequality gives us

$$\Pr[|Y_j - \mu| \ge \varepsilon\mu] \;\le\; 1/8\,.$$

Now, if fewer than $(s_2/2)$ of the Y_j's deviate by as much as $\varepsilon\mu$ from μ, then Y must be within $\varepsilon\mu$ of μ. So we upper bound the probability that this does not happen. Define s_2 indicator random variables I_j, where $I_j = 1$ iff $|Y_j - \mu| \ge \varepsilon\mu$, and let $W = \sum_{j=1}^{s_2} I_j$. Then $\mathrm{E}[W] \le s_2/8$. A standard Chernoff bound gives

$$\Pr\Big[|Y - \mu| \ge \varepsilon\mu\Big] \;\le\; \Pr\Big[W \ge \frac{s_2}{2}\Big] \;\le\; \left(\frac{e^3}{4^4}\right)^{s_2/8} \;=\; \left(\frac{e^3}{4^4}\right)^{\frac{1}{2}\lg(1/\delta)} \;\le\; \delta\,. \qquad \square$$

We use the following subroutine to compute a basic estimator X for the entropy norm F_H.

> ***Input stream***: $A = \langle a_1, a_2, \ldots, a_m \rangle$, where each $a_i \in \{1, \ldots, n\}$.
>
> 1 Choose p uniformly at random from $\{1, \ldots, m\}$.
> 2 Let $r = |\{q : a_q = a_p, p \le q \le m\}|$. Note that $r \ge 1$.
> 3 Let $X = m\big(r \lg r - (r-1)\lg(r-1)\big)$, with the convention that $0 \lg 0 = 0$.

Our algorithm for estimating the entropy norm outputs a final estimator based on this basic estimator, as described above. This gives us the following theorem.

Theorem 2.2. *If $F_H \geq m/\Delta$, for any $\Delta > 0$, the above one-pass algorithm can be implemented so that its output deviates from F_H by no more than εF_H with probability at least $1-\delta$, and so that it uses space*

$$O\left(\frac{\log(1/\delta)}{\varepsilon^2}\log m(\log m + \log n)\Delta\right).$$

In particular, taking Δ to be a constant, we have a polylogarithmic space algorithm that works on streams whose F_H is not "too small."

Proof. We first check that the expected value of X is indeed the desired quantity:

$$\begin{aligned}\mathrm{E}[X] &= \frac{m}{m}\sum_{v=1}^{n}\sum_{r=1}^{m_v}\big(r\lg r-(r-1)\lg(r-1)\big)\\ &= \sum_{v=1}^{n}(m_v\lg m_v - 0\lg 0) \;=\; F_H\,.\end{aligned}$$

The approximation guarantee of the algorithm now follows from Lemma 2.1. To bound the space usage, we must bound the variance $\mathrm{Var}[X]$ and for this we bound $\mathrm{E}[X^2]$. Let $f(r) := r\lg r$, with $f(0) := 0$, so that X can be expressed as $X = m(f(r)-f(r-1))$. Then

$$\begin{aligned}\mathrm{E}[X^2] &= m\sum_{v=1}^{n}\sum_{r=1}^{m_v}\big(f(r)-f(r-1)\big)^2\\ &\leq m\cdot\max_{1\leq r\leq m}\big(f(r)-f(r-1)\big)\cdot\sum_{v=1}^{n}\sum_{r=1}^{m_v}\big(f(r)-f(r-1)\big)\\ &\leq m\cdot\sup\big\{f'(x):\,x\in(0,m]\big\}\cdot F_H \qquad (2)\\ &= (\lg e+\lg m)\,mF_H \qquad (3)\\ &\leq (\lg e+\lg m)\,\Delta F_H^2\,,\end{aligned}$$

where (2) follows from the Mean Value Theorem.

Thus, $\mathrm{Var}[X]/E[X]^2 = O(\Delta\lg m)$. Moreover, the basic estimator can be implemented using space $O(\log m+\log n)$: $O(\log m)$ to count m and r, and $O(\log n)$ to store the value of a_p. Plugging these bounds into Lemma 2.1 yields the claimed upper bound on the space of our algorithm. □

Let F_0 denote the number of distinct items in the input stream (this notation deliberately coincides with that for frequency moments). Let $f(x) := x\lg x$ as used in the proof above. Observe that f is convex on $(0,\infty)$ whence, via Jensen's inequality, we obtain

$$F_H \;=\; \frac{F_0}{F_0}\sum_{v=1}^{n}f(m_v) \;\geq\; F_0 f\left(\frac{1}{F_0}\sum_{v=1}^{n}m_v\right) \;=\; m\lg\frac{m}{F_0}\,. \qquad (4)$$

Thus, if the input stream satisfies $m \geq 2F_0$ (or the simpler, but stronger, condition $m \geq 2n$), then we have $F_H \geq m$. As a direct corollary of Theorem 2.2 (for $\Delta = 1$) we obtain a $(1+\varepsilon)$-approximation algorithm for the entropy norm in space $O((\log(1/\delta)/\varepsilon^2)\log m(\log m + \log n))$. However, we can do slightly better.

Theorem 2.3. *If $m \geq 2F_0$ then the above one-pass, $(1+\varepsilon)$-approximation algorithm can be implemented in space*

$$O\left(\frac{\log(1/\delta)}{\varepsilon^2}\log m \log n\right)$$

without a priori knowledge of the stream length m.

Proof. We follow the proof of Theorem 2.2 up to the bound (3) to obtain $\mathrm{Var}[X] \leq (2\lg m)m F_H$, for m large enough. We now make the following claim

$$\frac{\lg m}{\lg(m/F_0)} \;\leq\; 2\max\{\lg F_0, 1\}\,. \tag{5}$$

Assuming the truth of this claim and using (4), we obtain

$$\mathrm{Var}[X] \;\leq\; (2\lg m)m F_H \;\leq\; \frac{2\lg m}{\lg(m/F_0)}F_H^2 \;\leq\; 4\max\{\lg F_0, 1\}F_H^2 \;\leq\; (4\lg n)F_H^2\,.$$

Plugging this into Lemma 2.1 and proceeding as before, we obtain the desired space upper bound. Note that we no longer need to know m before starting the algorithm, because the number of basic estimators used by the algorithm is now independent of m. Although maintaining each basic estimator seems, at first, to require prior knowledge of m, a careful implementation can avoid this, as shown by Alon et al [1].

We turn to proving our claim (5). We will need the assumption $m \geq 2F_0$. If $m \leq F_0^2$, then

$$\lg m \;\leq\; 2\lg F_0 \;=\; 2\lg F_0 \lg(2F_0/F_0) \;\leq\; 2\lg F_0 \lg(m/F_0)$$

and we are done. On the other hand, if $m \geq F_0^2$, then $F_0 \leq m^{1/2}$ so that

$$\lg(m/F_0) \;\geq\; \lg m - (1/2)\lg m \;=\; (1/2)\lg m$$

and we are done as well. □

Remark. Theorem 2.2 generalizes to estimating quantities of the form $\hat{\mu} = \sum_{v=1}^{n} \hat{f}(m_v)$, for any monotone increasing (on integer values), differentiable function $\hat{f}$ that satisfies $\hat{f}(0) = 0$. Assuming $\hat{\mu} \geq m/\Delta$, it gives us a one-pass $(1+\varepsilon)$-approximation algorithm that uses $\widetilde{O}(\hat{f}'(m)\Delta)$ space. For instance, this space usage is polylogarithmic in m if $\hat{f}(x) = x\,\mathrm{polylog}(x)$.

2.2 Lower Bound

The following lower bound shows that the algorithm of Theorem 2.2 is optimal, upto factors polylogarithmic in m and n.

Theorem 2.4. *Suppose Δ and c are integers with $4 \leq \Delta \leq o(m)$ and $0 \leq c \leq m/\Delta$. On input streams of size at most m, a randomized algorithm able to distinguish between $F_H \leq 2c$ and $F_H \geq c + 2m/\Delta$ must use space at least $\Omega(\Delta)$. In particular, the upper bound in Theorem 2.2 is tight in its dependence on Δ.*

Proof. We present a reduction from the classic problem of (two-party) Set Disjointness in communication complexity.

Suppose Alice has a subset X and Bob a subset Y of $\{1, 2, \ldots, \Delta - 1\}$, such that X and Y either are disjoint or intersect at exactly one point. Let us define the mapping

$$\phi : x \longmapsto \left\{ \frac{(m-2c)x}{\Delta} + i : i \in \mathbb{Z},\ 0 \leq i < \frac{m-2c}{\Delta} \right\}.$$

Alice creates a stream A by listing all elements in $\bigcup_{x \in X} \phi(x)$ and concatenating the c special elements $\Delta + 1, \ldots, \Delta + c$. Similarly, Bob creates a stream B by listing all elements in $\bigcup_{y \in Y} \phi(y)$ and concatenating the same c special elements $\Delta + 1, \ldots, \Delta + c$. Now, Alice can process her stream (with the hypothetical entropy norm estimation algorithm) and send over her memory contents to Bob, who can then finish the processing. Note that the length of the combined stream $A \circ B$ is at most $2c + |X \cup Y| \cdot ((m - 2c)/\Delta) \leq m$.

We now show that, based on the output of the algorithm, Alice and Bob can tell whether or not X and Y intersect. Since the set disjointness problem has communication complexity $\Omega(\Delta)$, we get the desired space lower bound.

Suppose X and Y are disjoint. Then the items in $A \circ B$ are all distinct except for the c special elements, which appear twice each. So $F_H(A \circ B) = c \cdot (2 \lg 2) = 2c$. Now suppose $X \cap Y = \{z\}$. Then the items in $A \circ B$ are all distinct except for the $(m-2c)/\Delta$ elements in $\phi(z)$ and the c special elements, each of which appears twice. So $F_H(A \circ B) = 2(c + (m-2c)/\Delta) \geq c + 2m/\Delta$, since $\Delta \geq 4$. □

Remark. Notice that the above theorem rules out even a polylogarithmic space *constant factor* approximation to F_H that can work on streams with "small" F_H. This can be seen by setting $\Delta = m^\gamma$ for some constant $\gamma > 0$.

3 Estimating the Empirical Entropy

We now turn to the estimation of the empirical entropy H of a data stream, defined as in equation (1): $H = \sum_{i=1}^{n} (m_i/m) \lg(m/m_i)$. Although H can be computed exactly from F_H, as shown in (1), a $(1 + \varepsilon)$-approximation of F_H can yield a poor estimate of H when H is small (sublinear in its maximum value, $\lg m$). We therefore present a different sublinear space, one-pass algorithm that directly computes entropy.

Our data structure takes a user parameter $\alpha > 0$, and consists of three components. The first (A1) is a sketch in the manner of Section 2, with basic estimator

$$X = m\left(\frac{r}{m}\lg\frac{m}{r} - \frac{r-1}{m}\lg\frac{m}{r-1}\right), \tag{6}$$

and a final estimator derived from this basic estimator using $s_1 = (8/\varepsilon^2)m^{2\alpha}\lg^2 m$ and $s_2 = 4\lg(1/\delta)$. The second component (A2) is an array of $m^{2\alpha}$ counters (each counting from 1 to m) used to keep exact counts of the first $m^{2\alpha}$ distinct items seen in the input stream. The third component (A3) is a Count-Min Sketch, as described by Cormode and Muthukrishnan [4], which we use to estimate k, defined to be the number of items in the stream that are *different* from the most frequent item; i.e., $k = m - \max\{m_i : 1 \le i \le n\}$. The algorithm itself works as follows. Recall that F_0 denotes the number of distinct items in the stream.

```
1 Maintain A1, A2, A3 as described above. When queried (or at end of input):
2 if F_0 ≤ m^{2α} then return exact H from A2.
3 else
4     let k̂ = estimate of k from A3.
5     if k̂ ≥ (1 − ε)m^{1−α} then return final estimator, Y, of A1.
6     else return (k̂ lg m)/m.
7 end
```

Theorem 3.1. *The above algorithm uses*

$$O\left(\frac{\log(1/\delta)}{\varepsilon^2}m^{2\alpha}\lg^2 m\right)$$

space and outputs a random variable Z that satisfies the following properties.

1. *If $k \le m^{2\alpha} - 1$, then $Z = H$.*
2. *If $k \ge m^{1-\alpha}$, then $\Pr\left[|Z - H| \ge \varepsilon H\right] \le \delta$.*
3. *Otherwise (i.e., if $m^{2\alpha} \le k < m^{1-\alpha}$), Z is a $(1/\alpha)$-approximation of H.*

Proof. The space bound is clear from the specifications of A1, A2 and A3, and Lemma 2.1. We now prove the three claimed properties of Z in sequence.

PROPERTY 1: This follows directly from the fact that $F_0 \le k + 1$.

PROPERTY 2: The Count-Min sketch guarantees that $\hat{k} \le k$ and, with probability at least $1 - \delta$, $\hat{k} \ge (1 - \varepsilon)k$. The condition in Property 2 therefore implies that $\hat{k} \ge (1 - \varepsilon)m^{1-\alpha}$, that is, $Z = Y$, with probability at least $1 - \delta$. Here we need the following lemma.

Lemma 3.2. *Given that the most frequent item in the input stream A has count $m-k$, the minimum entropy $H_{\min}$ is achieved when all the remaining k items are identical, and the maximum $H_{\max}$ is achieved when they are all distinct. Therefore,*

$$\begin{aligned} H_{\min} &= \frac{m-k}{m}\lg\frac{m}{m-k} + \frac{k}{m}\lg\frac{m}{k}\,, \quad \textit{and} \\ H_{\max} &= \frac{m-k}{m}\lg\frac{m}{m-k} + \frac{k}{m}\lg m\,. \end{aligned}$$

Proof. Consider a minimum-entropy stream $A_{\min}$ and suppose that, apart from its most frequent item, it has at least two other items with positive count. Without loss of generality, let $m_1 = m-k$ and $m_2, m_3 \geq 1$. Modify $A_{\min}$ to A' by letting $m'_2 = m_2 + m_3$ and $m'_3 = 0$, and keeping all other counts the same. Then

$$\begin{aligned} H(A') - H(A_{\min}) &= (\lg m - F_H(A')/m) - (\lg m - F_H(A_{\min})/m) \\ &= (F_H(A_{\min}) - F_H(A'))/m \\ &= m_2\lg m_2 + m_3\lg m_3 - (m_2+m_3)\lg(m_2+m_3) \\ &< 0\,, \end{aligned}$$

since $x\lg x$ is convex and monotone increasing (on integer values), giving us a contradiction. The proof of the maximum-entropy distribution is similar. □

Now, consider equation (6) and note that for any r, $|X| \leq \lg m$. Thus, if $\mathrm{E}[X] = H \geq 1$, then $\mathrm{Var}[X]/\mathrm{E}[X]^2 \leq E[X^2] \leq \lg^2 m$ and our choice of s_1 is sufficiently large to give us the desired $(1+\varepsilon)$-approximation, by Lemma 2.1. On the other hand, if $H < 1$, then $k < m/2$, by a simple argument similar to the proof of Lemma 3.2. Using the expression for $H_{\min}$ from Lemma 3.2, we then have

$$H_{\min} \;=\; \lg\frac{m}{m-k} + \frac{k}{m}\lg\frac{m-k}{k} \;\geq\; -\lg\left(1-\frac{k}{m}\right) \;\geq\; \frac{k}{m} \;\geq\; m^{-\alpha}\,,$$

which gives us $\mathrm{Var}[X]/\mathrm{E}[X]^2 \leq \mathrm{E}[X^2]/m^{-2\alpha} \leq (\lg^2 m)m^{2\alpha}$. Again, plugging this and our choice of s_1 into Lemma 2.1 gives us the desired $(1+\varepsilon)$-approximation.

PROPERTY 3: By assumption, $m^{2\alpha} \leq k < m^{1-\alpha}$. If $\hat{k} \geq (1-\varepsilon)m^{1-\alpha}$, then $Z = Y$ and the analysis proceeds as for Property 2. Otherwise, $Z = (\hat{k}\lg m)/m \leq (k\lg m)/m$. This time, again by Lemma 3.2, we have

$$H_{\min} \;\geq\; \frac{k}{m}\lg\frac{m}{k} \;\geq\; \frac{k}{m}\lg\left(m^{\alpha}\right) \;=\; \frac{\alpha k}{m}\lg m\,,$$

and

$$H_{\max} \;=\; \frac{m-k}{m}\lg\frac{m}{m-k} + \frac{k}{m}\lg m \;=\; \lg\frac{m}{m-k} + \frac{k}{m}\lg(m-k) \;\leq\; \frac{k}{m}\lg m + O\left(\frac{k}{m}\right)\,,$$

which, for large m, implies $H - o(H) \leq Z \leq H/\alpha$ and gives us Property 3. □

Corollary 3.3. *For $\alpha = 1/3$, the third case (Property 3) never occurs, so we have a $(1+\varepsilon)$-approximation in space $\widetilde{O}(m^{2/3})$.*

4 Conclusions

We have presented one-pass sublinear space algorithms for approximating the entropy norms as well as the empirical entropy. It will be of interest to study these problems on streams in presence of inserts and deletes.

References

[1] N. Alon, Y. Matias and M. Szegedy. The space complexity of approximating the frequency moments. *Proc. ACM STOC*, 20–29, 1996.

[2] B. Babcock, S. Babu, M. Datar, R. Motwani and J. Widom. Models and issues in data stream systems. *ACM PODS*, 2002, 1–16.

[3] D. Coppersmith and R. Kumar. An improved data stream algorithm for frequency moments. *ACM-SIAM SODA*, 151–156, 2004.

[4] G. Cormode and S. Muthukrishnan. An improved data stream summary: the count-min sketch and its applications. *J. Algorithms*, 55(1): 58–75, April 2005.

[5] C. Estan and G. Varghese. New directions in traffic measurement and accounting: Focusing on the elephants, ignoring the mice. *ACM Trans. Comput. Syst.*, 21(3): 270–313, 2003.

[6] P. Indyk and D. Woodruff. Optimal approximations of the frequency moments of data streams. *ACM STOC*, 202–208, 2005.

[7] S. Muthukrishnan. Data Streams: Algorithms and Applications. *Manuscript*, Available online at http://www.cs.rutgers.edu/˜muthu/stream-1-1.ps

[8] K. Xu, Z. Zhang, snd S. Bhattacharya. Profiling Internet Backbone Traffic: Behavior Models and Applications. To appear in *Proc. ACM SIGCOMM* 2005.