University of Wrocław: Algorithms for Big Data (Spring'22) 27/04/2022

Lecture 8: Sparse FFT

Lecturer: *Przemysław Uznański*

1 Sparse FFT - setting, motivation

Fourier transform: signal $\to$ frequencies.

Definition 1. *Assume $a = (a_0, \ldots, a_{n-1})$ is a signal. Let ω be n-th root of unity, that is $\omega = e^{\frac{2\pi}{n}i}$. Let F be such that $F_{ij} = \frac{1}{\sqrt{n}}\omega^{ij}$. Then $\hat{a} = Fa$ is a (Discrete) Fourier transform of a.*

DFT can be computed in $\mathcal{O}(n \log n)$ time. However, for some applications this time can already be prohibitive. Consider a following scenario (of signal compression):

- Take input signal a and compute $\hat{a}$.
- Let $\hat{a}_k$ be $\hat{a}$ with only k largest magnitude elements kept (rest is zeroed).
- Output $a_k = F^{-1}\hat{a}_k$.

If we consider complexity measure of Fourier support $fs(a) = \|\hat{a}\|_0$ that is number of non-zero Fourier coefficients, then actually there is

$$a_k = \arg \min_{x: fs(x) \leq k} \|a - x\|_2$$

(proof: exercise)

If we assume that a comes from real-life scenarios (photos, audio recording), then it should have only few "strong" frequencies, rest is noise. Since a_k has much simpler representation (namely, $\hat{a}_k$ which takes $\mathcal{O}(k \log n)$ bits), this is a lossy compression scheme.

How do we compute $\hat{a}_k$ efficiently? (Assumption is that we have random access to a, otherwise just *reading* the input would dominate the computation time.)

Simpler question: can we recover $\hat{a}$ if we know that $fs(a) = \|\hat{a}\|_0 \leq k$ (so there are only k non-zero frequencies)?

2 Sparse FFT - no noise [Kui92]

2.1 Recovering sparse signal

Assume wlog n is power of two.

Define $p_{d,\ell}(x) = \sum_{i: i \bmod 2^\ell = d} \hat{a}_i x^i$, and for a polynomial $p(x)$ we write $\|p\|^2 = \sum_i p_i^2$.

First trick is how to estimate $\|p_{d,\ell}\|_2^2$ with additive error. Given such blackbox, we can proceed:

1. Define $S_\ell = \{i : \|p_{i,\ell}\|^2 > 0\}$
2. Given S_ℓ, compute $S_{\ell+1}$: for each $d \in S_\ell$, test for $e \in \{d, d + 2^\ell\}$ whether $\|p_{e,\ell+1}\|^2 > 0$ and if so, add e to $S_{\ell+1}$.

3. $p_{i,\log n} = \hat{a}_i$

The idea is to start at level 0 and proceed. The size of each S_ℓ is bounded by k, thus the total number of steps (2) done is $\mathcal{O}(k \log n)$. Assume wlog n is power of two.

Plan of solution: Define $p_{d,\ell}(x) = \sum_{i : i \bmod 2^\ell = d} \hat{a}_i x^i$, and for short $p(x) = p_{0,0}(x) = \sum_i \hat{a}_i x^i$. and for a polynomial $f(x)$ we write for any polynomial $\|f\|_2^2 = \sum_i f_i^2$.

1. Define $S_\ell = \{i : \|p_{i,\ell}\|^2 > 0\}$
2. Given S_ℓ, compute $S_{\ell+1}$: for each $d \in S_\ell$, test for $e \in \{d, d + 2^\ell\}$ whether $\|p_{e,\ell+1}\|^2 > 0$ and if so, add e to $S_{\ell+1}$.
3. $p_{i,\log n}(1) = \hat{a}_i$

The idea is to start at level 0 and proceed. The size of each S_ℓ is bounded by k, thus the total number of steps (2) done is $\mathcal{O}(k \log n)$.

A few identities:

By inverse Fourier transform

$$p(\omega^t)/\sqrt{n} = \frac{1}{\sqrt{n}} \sum_i \hat{a}_i \omega^{it} = a_{-t}$$

By Parseval's theorem

$$\|p\|_2^2 = \sum_i \hat{a}_i^2 = \sum_i a_i^2 = \frac{1}{n} \sum_i |p(\omega^i)|^2$$

so

$$\|p_{d,\ell}\|_2^2 = \frac{1}{n} \sum_{i=0}^{n-1} |p_{d,\ell}(\omega^i)|^2$$

Additionally

$$\begin{aligned}
\frac{1}{2^\ell}\sqrt{n} \sum_{i=0}^{2^\ell - 1} a_{-(t+i\frac{n}{2^\ell})} \omega^{-di\frac{n}{2^\ell}} &= \frac{1}{2^\ell} \sum_{i=0}^{2^\ell - 1} p(\omega^{t+i\frac{n}{2^\ell}}) \omega^{-di\frac{n}{2^\ell}} \\
&= \frac{1}{2^\ell} \sum_{i=0}^{2^\ell - 1} \sum_{j=0}^{n-1} \hat{a}_j (\omega^{t+i\frac{n}{2^\ell}})^j \omega^{-di\frac{n}{2^\ell}} \\
&= \sum_{j=0}^{n-1} \hat{a}_j \omega^{tj} \frac{1}{2^\ell} \sum_{i=0}^{2^\ell - 1} \omega^{i(j-d)\frac{n}{2^\ell}} \\
&= \sum_{j=0}^{n-1} \hat{a}_j \omega^{tj} [j - d = 0 \bmod 2^\ell] \\
&= p_{d,\ell}(\omega^t)
\end{aligned}$$

Estimating sum via sampling: Lets say we have $a_1, \ldots, a_n$ such that $|a_i| \le H$. We can use $\mathcal{O}(H^2/\varepsilon^2 \cdot \log n)$ samples to obtain $\pm\varepsilon$ estimate of $\frac{1}{n}\sum_i a_i$, or equivalently $\pm n\varepsilon$ estimate of $\sum_i a_i$, with proof via Hoeffding bound.

Solution: Denote $L = \max|\hat{a}_i|$ and $H = \max|p(\omega^t)|$. Observe that $p_{d,\ell}(\omega^t) \le H$. Since $p(\omega^t) = \sum_j \hat{a}_j \omega^{-tj}$, we have $H \le kL$.

1. Estimate $A_{d,\ell,t} \approx p_{d,\ell}(\omega^t)$ using $\mathcal{O}(H^4 \log n)$ samples of $a_{-t-i\frac{n}{2^\ell}}\omega^{-di\frac{n}{2^\ell}}$, up to error $\pm\frac{1}{16H}$.
2. $|A_{d,\ell,t}|^2 - |p_{d,\ell}(\omega^t)|^2 \le (|A_{d,\ell,t}| - |p_{d,\ell}(\omega^t)|)(|A_{d,\ell,t}| + |p_{d,\ell}(\omega^t)|) \le \frac{1}{16H} \cdot 2H \le \frac{1}{4}$
3. $\|p_{d,\ell}\|_2^2 = \frac{1}{n}\sum_{t=0}^{n-1}|p_{d,\ell}(\omega^t)|^2 = \frac{1}{n}\sum_{t=0}^{n-1}(|A_{d,\ell,t}|^2 \pm \frac{1}{4}) = (\sum_{t=0}^{n-1}|A_{d,\ell,t}|^2) \pm 1/4$
4. We have $|A_{d,\ell,t}|^2 = \mathcal{O}(H^2)$.
5. We sample $\mathcal{O}(H^4 \log n)$ of $|A_{d,\ell,t}|^2$ to estimate $\|p_{d,\ell}\|_2^2$ up to $\pm\frac{1}{2}$.
6. In total $\mathcal{O}(H^8 \log^2 n) = \mathcal{O}(k^8 L^8 \log^2 n)$ samples to compute such estimate.

Applying to our tree-traversal, we get $\mathcal{O}(k^9 L^8 \log^3 n)$ complexity. Once we have the indices of non-zero coefficients, we extract the exact values:

$$\hat{a}_d = p_{d,\log n}(\omega^0) = \frac{1}{n}\sum_{i=0}^{n-1} a_i \omega^{di}$$

via sampling.

3 Another algorithm, noisy case [LLL16] [EÇ18]

3.1 $k = 1$

There is some heavy $\hat{a}_u$ such that $\sum_{u' \ne u}|\hat{a}_{u'}|^2 \le \varepsilon|\hat{a}_u|^2$, for some small constant ε.

Idea: extract u bit-by-bit.

No noise: If $u = 2v + b_0$ for $b_0 \in \{0, 1\}$, then $a_{n/2} = \frac{1}{\sqrt{n}}\hat{a}_u\omega^{un/2} = \frac{1}{\sqrt{n}}\hat{a}_u\omega^{vn+b_0n/2} = \frac{1}{\sqrt{n}}\hat{a}_u\omega^{b_0n/2} = \frac{1}{\sqrt{n}}\hat{a}_u(-1)^{b_0}$. Then, $a_0 = \frac{1}{\sqrt{n}}\hat{a}_u$. Thus we can use following test:

$$b_0 = 0 \qquad \text{iff} \qquad |a_0 - a_{n/2}| \le |a_0 + a_{n/2}|$$

How to test for older bits? Assume wlog that $b_0 = 0$, since if $b_0 = 1$, we can always consider signal a' defined as $a'_j = a_j \cdot \omega^j$, where $\hat{a}'_j = \hat{a}_{j-1}$. So $u = 4v' + 2b_1$, where $b_1 \in \{0, 1\}$. We then observe that $a_{n/4} = \frac{1}{\sqrt{n}}\hat{a}_u(-1)^{b_1}$, so the test is then

$$b_1 = 0 \qquad \text{iff} \qquad |a_0 - a_{n/4}| \le |a_0 + a_{n/4}|$$

So we can proceed with all the bits in this manner.

Noisy case: Consider test for bit 0. If the noise is concentrated around a_0 and $a_{n/2}$, then such test fails. But we know that on average the noise is small. Thus we replace the test with a randomized one: pick $0 \le r < n$ at random, and test:

$$b_0 = 0 \qquad \text{iff} \qquad |a_r - a_{r+n/2}| \le |a_r + a_{r+n/2}|$$

(we can do many tests and pick majority vote) and in general

$$b_{i-1} = 0 \qquad \text{iff} \qquad |a_r - a_{r+n/2^i}| \le |a_r + a_{r+n/2^i}|$$

of course assuming $b_0 = b_1 = .. = b_{i-2} = 0$, and changing the signal accordingly.

Why does it work?

Let $\hat{a}'$ be the output. We show that with ppb at least 3/4 there is $\|\hat{a} - \hat{a}'\|_2 \le \varepsilon \|\hat{a} - \hat{a}^{(1)}\|_2$, where $\hat{a}^{(1)}$ is the top coefficient, so $\hat{a} - \hat{a}^{(1)}$ is the noise.

We rewrite

$$a_j = \frac{1}{\sqrt{n}} \hat{a}_u \omega^{uj} + \frac{1}{\sqrt{n}} \sum_{u' \neq u} \hat{a}_{u'} \omega^{u'j} = \frac{1}{\sqrt{n}} \hat{a}_u \omega^{uj} + \mu_j$$

so

$$a = F^{-1} \hat{a}^{(1)} + \mu$$

Looking at the error

$$\sum_{j=0}^{n-1} |\mu_j|^2 = \|\mu\|_2^2 = \|F^{-1}(\hat{a} - \hat{a}^{(1)})\|_2^2 = \|\hat{a} - \hat{a}^{(1)}\|_2^2 = \sum_{u' \neq u} |\hat{a}_{u'}|^2$$

Algorithm at single step compares $|a_k - a_\ell|$ vs $|a_k + a_\ell|$. We have

$$a_k - a_\ell = \frac{1}{\sqrt{n}} \hat{a}_u (\omega^{uk} - \omega^{u\ell}) + (\mu_k - \mu_\ell)$$

$$a_k + a_\ell = \frac{1}{\sqrt{n}} \hat{a}_u (\omega^{uk} + \omega^{u\ell}) + (\mu_k + \mu_\ell)$$

We know that $|\omega^{uk} \pm \omega^{u\ell}| \in \{0, 2\}$ so for the comparison to be done correctly, it is enough that $|\mu_k - \mu_\ell| + |\mu_k + \mu_\ell| \le 2 \frac{1}{\sqrt{n}} \hat{a}_u$, so $|\mu_k| + |\mu_\ell| \le \frac{1}{\sqrt{n}} \hat{a}_u$.

We now have, since each index is picked with a random shift,

$$\mathbb{E}[|\mu_k|^2] = \frac{1}{n} \sum_{j=0}^{n-1} |\mu_j|^2 = \frac{1}{n} \sum_{u' \neq u} |\hat{a}_{u'}|^2 \le \frac{1}{n} \varepsilon |\hat{a}_u|^2$$

so

$$\Pr[|\mu_k| \le \frac{1}{2\sqrt{n}} |\hat{a}_u|] \le \frac{\frac{1}{n} \varepsilon |\hat{a}_u|^2}{\frac{1}{4n} |\hat{a}_u|^2} = \frac{\varepsilon}{4}$$

So picking $\varepsilon = 1/2$ gives us by union bound 1/4 ppb of success.

Now the trick is to amplify the ppb by repeating each test $\mathcal{O}(\log \log n)$ times and do the majority vote. This amplifies the ppb to $1/(4 \log n)$, so by union bound the whole procedure is ok with 3/4 ppb.

Not covered: how to extract the value of $\hat{a}_u$.

References

[EÇ18] Esra Sengun Ermeydan and Ilyas Çankaya. Sparse fast fourier transform for exactly sparse signals and signals with additive gaussian noise. *Signal, Image and Video Processing*, 12(3):445–452, 2018.

[Kui92] W. Kuich, editor. *Randomized interpolation and approximation of sparse polynomials: Preliminary version*, Berlin, Heidelberg, 1992. Springer Berlin Heidelberg.

[LLL16] Pierre-David Letourneau, M. Harper Langston, and Richard Lethin. A sparse multi-dimensional fast fourier transform with stability to noise in the context of image processing and change detection. In *2016 IEEE High Performance Extreme Computing Conference, HPEC 2016, Waltham, MA, USA, September 13-15, 2016*, pages 1–6. IEEE, 2016.