Graphs and Networks Lecture 11

Personal PageRank and Spilling Paint

Daniel A. Spielman October 7, 2010

11.1 Overview

- These lecture notes are not complete.
- The paint spilling metaphor is due to Berkhin [Ber06].

11.2 PageRank

We have all encountered the PageRank algorithm: it is how Google got started ranking web pages. It involves random walks in directed graphs. We are going to talk more about random walks in directed graphs in a later lecture. But, I need to give you the introduction now.

PageRank considers a process that with probability α jumps to a uniformly random vertex of a graph, and with probability $1-\alpha$ follows a random edge out of the present node. The PageRank vector is the steady-state distributuion of this process. That is, if we let $\boldsymbol{W}$ be the walk matrix of the directed graph (you can figure out how to define it), the PageRank vector $\boldsymbol{p}$ will satisfy

$$\boldsymbol{p} = \alpha \frac{1}{n}\mathbf{1} + (1-\alpha) W \boldsymbol{p}.$$

We are going to consider a variation of the PageRank vector called the *personal PageRank vector*. Where PageRank measures the importance of nodes overall, the personal PageRank vector measures the importance of nodes with respect to a give node u. It does this by modifying the walk so that with probability α it jumps back to u, rather than to a random node. We denote the vector that satisfies this equation by $\boldsymbol{p}_u$, and note that it must satisfy the equation

$$\boldsymbol{p}_u = \alpha \chi_u + (1-\alpha)\, \boldsymbol{W} \boldsymbol{p},$$

where χ_u is the elementary unit vector in the direction of vertex u.

For today, we will just consider these vectors in *undirected* graphs.

Let's begin by showing that the vectors $\boldsymbol{p}_u$ actuall exist. By maniuplating the equation for $\boldsymbol{p}_u$, we derive

$$\begin{aligned} \boldsymbol{p}_u - (1-\alpha)\, \boldsymbol{W} \boldsymbol{p}_u &= \alpha \chi_u \\ [\boldsymbol{I} - (1-\alpha)\, \boldsymbol{W}]\, \boldsymbol{p}_u &= \alpha \chi_u \\ \boldsymbol{p}_u &= [\boldsymbol{I} - (1-\alpha)\, \boldsymbol{W}]^{-1}\, \alpha \chi_u. \end{aligned}$$

Before we write this, we should be sure that that the inverse exists. We know that it does because all eigenvalues of $\boldsymbol{W}$ lie between -1 and 1, so all eigenvalues of $\boldsymbol{I} - (1-\alpha)\boldsymbol{W}$ are at least α.

11.3 Spilling Paint in a Graph

Just as ordinary random walks were related to a diffusion process, the PageRank random walks are as well. However, we should think of this diffusion process as diffusing paint in a graph. The main characterisic of paint is that it dries.

In our model, we will say that at every time step an α fraction of the paint at each vertex dries in place. As for the wet paint, we assume that half stays where it is and the other half is distributed equally among its neighbors. So, we will need to keep track of two quantities, the amount of wet paint and the amount of dried paint. We will let $\boldsymbol{s} : V \to \mathbb{R}^{\geq 0}$ be the vector the records how much paint has become *stuck* at each vertex, and we will let $\boldsymbol{r} : V \to \mathbb{R}^{\geq 0}$ indicate how much wet paint *remains* at each vertex. At time zero, we set $\boldsymbol{r}^0 = \chi_u$. These vectors now evolve according to the equations

$$\begin{aligned}\boldsymbol{s}^{t+1} &= \boldsymbol{s}^t + \alpha \boldsymbol{r}^t \\ \boldsymbol{r}^{t+1} &= (1-\alpha)\,\widehat{\boldsymbol{W}}\,\boldsymbol{r}^t.\end{aligned}$$

We will be interested in where the paint is stuck in the end. We could denote this by $\boldsymbol{s}^\infty$. We derive the following equation for $\boldsymbol{s}^\infty$:

$$\boldsymbol{s}^\infty = \alpha \sum_{t \geq 0} \boldsymbol{r}^t = \alpha \sum_{t \geq 0} (1-\alpha)^t \widehat{\boldsymbol{W}}^t \boldsymbol{r}^0 = \alpha \sum_{t \geq 0} (1-\alpha)^t \widehat{\boldsymbol{W}}^t \chi_u.$$

We will now see that this is the same equation that the personal PageRank vector satisfies, up to scaling and with a slightly different α. The reason for the different α is that here we used the lazy walk matrix, whereas in personal PageRank we used the ordinary walk matrix. We will use a fact that is just as true of diagonalizable matrices as it is of real numbers:

$$(\boldsymbol{I} - \boldsymbol{X})^{-1} = \boldsymbol{I} + \boldsymbol{X} + \boldsymbol{X}^2 + \boldsymbol{X}^3 + \cdots,$$

provided that all eigenvalues of $\boldsymbol{X}$ have absolute value less than 1.

So,

$$\boldsymbol{p}_u = \alpha \sum_{t \geq 0} (1-\alpha)^t \boldsymbol{W}^t \chi_u.$$

To see that the difference between using $\boldsymbol{W}$ and $\widehat{\boldsymbol{W}}$ is just a change in α, consider the equation

which we now know $\boldsymbol{s}^\infty$ satisfies:

$$\begin{aligned}\boldsymbol{s}^\infty &= \alpha\left(\boldsymbol{I} - (1-\alpha)\,\widehat{\boldsymbol{W}}\right)^{-1}\chi_u \\ &= \alpha\left(\frac{1+\alpha}{2}\boldsymbol{I} - \frac{1-\alpha}{2}\,\boldsymbol{W}\right)^{-1}\chi_u \\ &= \frac{2\alpha}{1+\alpha}\left(\boldsymbol{I} - \frac{1-\alpha}{1+\alpha}\,\boldsymbol{W}\right)^{-1}\chi_u \\ &= \beta\left(\boldsymbol{I} - (1-\beta)\,\boldsymbol{W}\right)^{-1}\chi_u,\end{aligned}$$

where

$$\beta = \frac{2\alpha}{1+\alpha}.$$

11.4 Local Updates

From the discussion so far, we can see two obvious ways of computing the vectors $\boldsymbol{p}_u$: either by solving a linear system or by simulating the paint diffusion process. It turns out that there is a very nice way of simulating the paint diffusion process. It does not need to be done globally through the equations we derived. Rather, we can arbitrarily pick vertices of the graph, proclaim an α fraction of the wet paint at those vertices dry, and then push the wet paint to neighbors as appropriate.

That is, we can ignore time, and do this by a completely asynchronous process. Since we will ignore time, let $\boldsymbol{s}$ be the vector of dried paint and let $\boldsymbol{r}$ be the vector of wet paint. Let's denote by $\boldsymbol{p}_{\boldsymbol{s},\boldsymbol{r}}$ the vector that we will eventually compute:

$$\boldsymbol{p}_{\boldsymbol{s},\boldsymbol{r}} = \boldsymbol{s} + \alpha\sum_{t\geq 0}(1-\alpha)^t\,\boldsymbol{W}^t\boldsymbol{r} = \boldsymbol{s} + \alpha\left(\boldsymbol{I} - (1-\alpha)\,\boldsymbol{W}\right)^{-1}\boldsymbol{r}.$$

Remark I'm changing from lazy to ordinary walk here as we know it won't make any difference, and it saves ink.

I am claiming that we can now update $\boldsymbol{s}$ and $\boldsymbol{r}$ as follows. Pick an arbitrary vertex u. Now, create the new vectors $\boldsymbol{s}'$ and $\boldsymbol{r}'$ by the rules

$$\begin{aligned}\boldsymbol{s}'(u) &= \boldsymbol{s}(u) + \alpha\boldsymbol{r}(u) \\ \boldsymbol{p}'(u) &= 0 \\ \boldsymbol{p}'(v) &= \boldsymbol{p}(v) + \frac{1-\alpha}{d(u)}\boldsymbol{p}(u), \text{for every neighbor } v \text{ of } u.\end{aligned}$$

Lemma 11.4.1.

$$\boldsymbol{p}_{\boldsymbol{s}',\boldsymbol{r}'} = \boldsymbol{p}_{\boldsymbol{s},\boldsymbol{r}}.$$

Proof. In vector notation,

$$\begin{aligned}\boldsymbol{s}' &= \boldsymbol{s} + \alpha\boldsymbol{p}(u)\chi_u, \quad \text{and,} \\ \boldsymbol{r}' &= \boldsymbol{r} - \boldsymbol{p}(u)\chi(u) + (1-\alpha)\boldsymbol{p}(u)\,\boldsymbol{W}\chi_u.\end{aligned}$$

So,

$$\begin{aligned}
\boldsymbol{p}_{\boldsymbol{s}',\boldsymbol{r}'} &= \boldsymbol{s}' + \alpha \sum_{t \geq 0} (1-\alpha)^t \boldsymbol{W}^t \boldsymbol{r}' \\
&= \boldsymbol{s} + \alpha \boldsymbol{p}(u)\chi_u + \alpha \sum_{t \geq 0} (1-\alpha)^t \boldsymbol{W}^t \boldsymbol{r} - \alpha \sum_{t \geq 0} (1-\alpha)^t \boldsymbol{W}^t \boldsymbol{p}(u)\chi_u + \alpha \sum_{t \geq 0} (1-\alpha)^t \boldsymbol{W}^t (1-\alpha)\boldsymbol{W}\chi_u \\
&= \boldsymbol{p}_{\boldsymbol{s},\boldsymbol{r}} + \alpha \boldsymbol{p}(u)\chi_u - \alpha \sum_{t \geq 0} (1-\alpha)^t \boldsymbol{W}^t \boldsymbol{p}(u)\chi_u + \alpha \sum_{t \geq 0} (1-\alpha)^t \boldsymbol{W}^t (1-\alpha)\boldsymbol{W}\chi_u \\
&= \boldsymbol{p}_{\boldsymbol{s},\boldsymbol{r}} + \alpha \boldsymbol{p}(u) \left(\chi_u + \sum_{t \geq 1} (1-\alpha)^t \boldsymbol{W}^t \chi_u - \sum_{t \geq 0} (1-\alpha)^t \boldsymbol{W}^t \chi_u \right) \\
&= \boldsymbol{p}_{\boldsymbol{s},\boldsymbol{r}}.
\end{aligned}$$

□

This led to the idea of computing *approximate PageRank vectors*. The idea behind these is to always pick the vertex for which $\boldsymbol{r}(u)$ is largest, and then distribute the paint from this vertex. Once there is very little paint at every vertex, we stop the process. In particular, we choose some threshold ϵ, and don't bother to process a vertex if it satisfies

$$\boldsymbol{r}(u) \leq \epsilon d(u).$$

Under this situation, one can show that the process will stop within $1/\epsilon\alpha$ iterations.

This leads to a very interesting notion of how one should explore a graph. If you asked me 20 years ago how one should explore a graph from a vertex u, I would have given the obvious answer: "Breadth First Search". But, for graphs with low diameter, as we now know many are, this is not so useful. I prefer this way of exploring a graph. We only explore nodes when we process them for the first time. Still, this can be improved. Unlike in breadth first search, this process could involve a lot of computation that does not lead to the exploration of new vertices. For example, you can see this if you simulate it on a path graph.

Question Can we improve on this exploration process in a reasonable way?

One improvement has been provided by Andersen and Peres [AP09]. But, I believe we can do better.

11.5 Personal PageRank and Conductance

Andersen, Chung and Lang [ACL06] show that we can use personal PageRank vectors to find sets of low conductance, if we start from a random vector in such a set. Actually, they do this for approximate personal PageRank vectors. This is particularly nice because the number of vertices the algorithm touches is actually proportional to the size of the set that it outputs. So, if there is

a small set of low conductance then the algorithm will run very quickly. This is desirable in very large graphs.

Their analysis also uses the holistic approach of Lovàsz and Simonovits. I will go through a simpler analysis that just examines the personal PageRank vector, without the approximation. This analysis mostly comes from the paper of Andersen and Chung [AC07]

From the vector $\boldsymbol{p}_v$, we derive the vector $\boldsymbol{q}_v$:

$$\boldsymbol{q}_v(u) = \frac{\boldsymbol{p}_v(u)}{d(u)}.$$

We now assume without loss of generality that the vertices are numbered so that

$$\boldsymbol{q}_v(1) \geq \boldsymbol{q}_v(2) \geq \cdots \geq \boldsymbol{q}_v(n).$$

Let S_k then be the set of vertices $\{1, \ldots, k\}$.

We will now make two observations about the vector $\boldsymbol{q}_v$. The first says that it drops slowly if α is small. The second says that it drops slowly if all of the sets S_j have high conductance. We will eventually show that if v is chosen at random in a set of small conductance, then $\boldsymbol{q}_v$ in unlikely to drop slowly. Thus, in this case, one of the sets S_j must have low conductance.

Lemma 11.5.1. *For every k,*

$$\sum_{i \leq k < j} \boldsymbol{q}(i) - \boldsymbol{q}(j) \leq \alpha.$$

Proof. We have

$$\begin{aligned}
\chi_S^T \boldsymbol{W} \boldsymbol{p} &= \sum_{(u,v) \in E, u \in S, v \in V} \boldsymbol{W}(u,v) \boldsymbol{p}(v) \\
&= \sum_{(u,v) \in E, u \in S, v \in S} \boldsymbol{q}(v) + \sum_{(u,v) \in E, u \in S, v \notin S} \boldsymbol{q}(v) \\
&= \sum_{v \in S} \boldsymbol{p}(v) - \sum_{(u,v) \in E, u \notin S, v \in S} \boldsymbol{q}(v) + \sum_{(u,v) \in E, u \in S, v \notin S} \boldsymbol{q}(v) \\
&= \chi_S^T \boldsymbol{p} - \sum_{(u,v) \in E, u \in S, v \notin S} \boldsymbol{q}(u) - \boldsymbol{q}(v).
\end{aligned}$$

On the other hand,

$$\boldsymbol{W} \boldsymbol{p} = (1-\alpha)^{-1} (\boldsymbol{p} - \alpha \chi_u) \geq \boldsymbol{p} - \alpha \chi_u,$$

and so

$$\chi_S^T \boldsymbol{W} \boldsymbol{p} \geq \chi_S^T \boldsymbol{p} - \alpha.$$

Together, these inequalities imply

$$\sum_{(u,v) \in E, u \in S, v \notin S} \boldsymbol{q}(u) - \boldsymbol{q}(v) \leq \alpha.$$

□

Lemma 11.5.2. *If $\phi(S_j) \geq 2\theta$, then there exists a $k > j$ such that*

$$d(S_k) \geq (1+\theta)d(S_j) \quad \textit{and} \quad \boldsymbol{q}(k) \geq \boldsymbol{q}(j) - \frac{\alpha}{\theta d(S_j)}.$$

Proof. Let k be the least integer such that $d(S_k) \geq (1+\theta)d(S_j)$. As at least $2\theta d(S_j)$ edges leave S_j and there are at most $\theta d(S_j)$ sockets between S_j and S_k, we know that at least $\theta d(S_j)$ of the edges leaving S_j must go past S_k. As $\boldsymbol{q}$ is a decreasing function, this tells us that

$$\sum_{(a,b)\in E: a\leq j, b\geq k} \boldsymbol{q}(a) - \boldsymbol{q}(b) \geq \theta d(S_j)(\boldsymbol{q}(j) - \boldsymbol{q}(k)).$$

The lemma now follows from Lemma 11.5.1, which tells us that this sum is at most α. □

Lemma 11.5.3. *Assume that $\phi(S_j) \geq 2\theta$ for all j such that $d(S_j) \leq 2m/3$. Let h be the least integer such that $d(S_h) \geq 2m/3$. Then, for every $i \leq h$*

$$\boldsymbol{q}(h) \geq \boldsymbol{q}(i) - \frac{2\alpha}{\theta^2 d(S_i)}.$$

Proof. We apply Lemma 11.5.2, starting at i, until we reach a $k \geq h$. We then get that

$$\boldsymbol{q}(h) \geq \boldsymbol{q}(i) - \frac{\alpha}{\theta d(S_i)} - \frac{\alpha}{\theta(1+\theta)d(S_i)} - \frac{\alpha}{\theta(1+\theta)^2 d(S_i)} \cdots$$

As

$$1 + \frac{1}{1+\theta} + \frac{1}{(1+\theta)^2} + \frac{1}{(1+\theta)^3} + \cdots \leq \frac{1+\theta}{\theta},$$

we get

$$\boldsymbol{q}(h) \geq \boldsymbol{q}(i) - \frac{\alpha}{\theta d(S_i)} \frac{1+\theta}{\theta} \geq \boldsymbol{q}(i) - \frac{2\alpha}{\theta^2 d(S_i)}.$$

□

We will show that if v is chosen according to degree inside a set of low conductance, then one of the sets S_k will probably have low conductance as well, at least for an appropriate choice of α. The parameter α will play the role of the reciprocal of time in a random walk. We begin by pointing out that if α is sufficiently large, then most of the paint will stay inside a set of low conductance.

For our analysis, we will find it convenient to extend the notion of personal PageRank vectors to start at a distribution over vertices. So, if $\boldsymbol{a}$ is a probability vector, we set

$$\boldsymbol{p}_{\boldsymbol{a}} \stackrel{\text{def}}{=} \sum_v \boldsymbol{a}(v)\boldsymbol{p}_v.$$

Lemma 11.5.4. *Let S be any set of vertices and let $\boldsymbol{\pi}_S$ be the distribution on S according to degree. Then*

$$\chi_{V-S}^T \boldsymbol{p}_{\boldsymbol{\pi}_S} \leq \phi(S)\frac{1+\alpha}{\alpha}.$$

Proof. Using the result on the problem set, we gte

$$\begin{aligned}
\chi_{V-S}^T \boldsymbol{p}_{\boldsymbol{\pi}_S} &= \alpha \sum_{t \geq 0} (1-\alpha)^t \chi_{V-S}^T \boldsymbol{W}^t \boldsymbol{\pi}_S \\
&\leq \alpha \sum_{t \geq 0} (1-\alpha)^t t \phi(S) \\
&= \alpha \phi(S) \sum_{t \geq 0} (1-\alpha)^t t \\
&= \alpha \phi(S) \frac{1+\alpha}{\alpha^2} \\
&= \phi(S) \frac{1+\alpha}{\alpha}.
\end{aligned}$$

□

We will apply this lemma when S is small set of low conductance. In this case, we will see that most of the mass lies inside S. This tells us that there is a small i for which $\boldsymbol{q}(i)$ is relatively large. We then use Lemma 11.5.3 to show that $\boldsymbol{q}(h)$ is almost as large as $\boldsymbol{q}(i)$. But, this will give an absurdity as the total probability mass is at least $d(S_h)\boldsymbol{q}(h)$, which will then be more than 1.

The question is, how large can I force $\boldsymbol{q}(i)$ to be? Assume that we have chosen α so that

$$\phi(S) \frac{1+\alpha}{\alpha} < 1/3.$$

Lemma 11.5.5. *There is an i such that $d(S_i) \leq d(S)$ and*

$$\boldsymbol{q}(i) \geq \frac{2/3}{d(S_i) H(2m)},$$

where

$$H(2m) \stackrel{\text{def}}{=} \sum_{j=1}^{2m} \frac{1}{j} \approx \ln(2m).$$

To check this, just assume the opposite and see that it gives $\boldsymbol{q}(S) < 2/3$.

Now, define $\gamma = H(2m)$. Assume that

$$d(S) \leq \frac{2m}{9\gamma},$$

and recall that $d(V) = 2m$.

Assume that every set S_j has conductance at least

$$\sqrt{6\alpha\gamma} \stackrel{\text{def}}{=} \theta.$$

We then find that

$$\boldsymbol{q}(h) \geq \boldsymbol{q}(i) - \frac{2\alpha}{\theta^2 d(S_i)} = \boldsymbol{q}(i) - \frac{2\alpha}{6\alpha\gamma d(S_i)} \geq \frac{2/3}{d(S_i)\gamma} \frac{1}{2} = \frac{1}{3d(S_i)\gamma}.$$

This gives

$$h\boldsymbol{q}(h) \geq \frac{1}{3}\frac{2m}{3d(S_i)\gamma} > 1,$$

given our assumption on $d(S)$. As this is a contradiction, we know that we must have found a set S_j of conductance less than θ, which is approximately

$$O(\sqrt{\phi(S)\log m}).$$

References

[AC07] Reid Andersen and Fan Chung. Detecting sharp drops in pagerank and a simplified local partitioning algorithm. In Jin-Yi Cai, S. Cooper, and Hong Zhu, editors, *Theory and Applications of Models of Computation*, volume 4484 of *Lecture Notes in Computer Science*, pages 1–12. Springer Berlin / Heidelberg, 2007.

[ACL06] Reid Andersen, Fan Chung, and Kevin Lang. Local graph partitioning using pagerank vectors. In *FOCS '06: Proceedings of the 47th Annual IEEE Symposium on Foundations of Computer Science*, pages 475–486, Washington, DC, USA, 2006. IEEE Computer Society.

[AP09] Reid Andersen and Yuval Peres. Finding sparse cuts locally using evolving sets. In *STOC '09: Proceedings of the 41st annual ACM symposium on Theory of computing*, pages 235–244, New York, NY, USA, 2009. ACM.

[Ber06] Pavel Berkhin. Bookmark-coloring algorithm for personalized pagerank computing. *Internet Mathematics*, 3(1):41–62, 2006.