Graph Mining
SD212
3. PageRank

Thomas Bonald
Institut Polytechnique de Paris

2019 – 2020

These lecture notes introduce PageRank, originally proposed for the Web [1], and related metrics to rank the nodes of a graph in terms of frequency of visits by a random walk.

1 Notation

Let $G = (V, E)$ be a directed graph of n nodes and m edges, with adjacency matrix A. Let $d^+ = A1$ and $d^- = A^T 1$ be the vectors of out-degrees and in-degrees. We say that node i is a sink if $d_i^+ = 0$. Unless otherwise specified, we assume that the graph G has no sink, i.e., the vector d^+ is positive.

2 Random walk

Consider a random walk in the graph G with a probability of moving from node i to node j equal to A_{ij}/d_i^+. Let $X_0, X_1, X_2, \dots$ be the sequence of nodes visited by the random walk. This defines an irreducible Markov chain on $\{1, \dots, n\}$ with transition matrix $P = D^{-1}A$, where $D = \text{diag}(d^+)$. Let π_t be the distribution of X_t, expressed as a row vector. We have for all $t \geq 1$:

$$\pi_t = \pi_{t-1} P, \tag{1}$$

so that $\pi_t = \pi_0 P^t$, where π_0 is the initial distribution. If the graph is strongly connected and aperiodic (that is, the largest common divisor of the cycle lengths is equal to 1), the following limit exists and is unique:

$$\pi = \lim_{t \to +\infty} \pi_t. \tag{2}$$

This is the stationary distribution of the random walk, which is the unique solution to the balance equations:

$$\pi = \pi P. \tag{3}$$

In particular, π is the unique left eigenvector of P for the eigenvalue 1 such that $\pi 1 = 1$ (observe that $P1 = 1$, that is, 1 is the corresponding right eigenvector). The vector π gives the frequency of visits of the random walk to each node, and as such provides a natural ranking of the nodes. In general, π cannot be computed exactly but, in view of (1)–(2), can be approximated by successive matrix-vector multiplications. It is independent of the initial distribution π_0.

Remark 1 *It can be shown that the sequence π_t converges to π at an exponential rate equal to the modulus of the second largest eigenvalue of P.*

Remark 2 *If the graph is not strongly connected, the limit* (2) *exists but depends on the initial distribution. In particular, the distribution over the connected components of the graph remains constant.*

Undirected graphs. If the graph is undirected, it can be easily verified that $\pi \propto d$, with $d = d^+ = d^-$: the frequency of visits to a node is proportional to its degree. In particular, there is no need to "solve" (3) in this case, as the solution is explicit.

Weighted graphs. If the graph is weighted, each edge is assigned a positive weight corresponding to its strength. The results apply in the same manner, with a probability of moving from node i to node j proportional to the weight of edge $i \to j$.

3 PageRank

If the graph G has sinks, the random walk is no longer defined. A natural approach consists in letting the random walk jump to any node chosen uniformly at random in V (in particular, the random walk stays in the same node with probability $1/n$). The transition matrix becomes:

$$P_{ij} = \begin{cases} \frac{A_{ij}}{d_i^+} & \text{if } d_i^+ > 0, \\ \frac{1}{n} & \text{otherwise.} \end{cases} \tag{4}$$

Another issue concerns the presence of connected components. The random walk may be trapped in a small clique for instance, giving to each node of the clique a ranking much higher than expected. The solution adopted by PageRank is to let the random walk restart with some fixed probability. Specifically, for some parameter $\alpha \in (0, 1)$, the random walk continues along the edges of the graph with probability α and restarts from some node chosen uniformly at random in V with probability $1 - \alpha$. The corresponding transition matrix is:

$$P^{(\alpha)} = \alpha P + (1 - \alpha)\frac{11^T}{n},$$

with P given by (4). The corresponding stationary distribution $\pi^{(\alpha)}$, called the PageRank vector, satisfies:

$$\pi^{(\alpha)} = \alpha \pi^{(\alpha)} P + (1 - \alpha)\frac{1^T}{n}. \tag{5}$$

It can be computed either by inverting the matrix $I - \alpha P$ or through iterations, starting from some arbitrary distribution (e.g., uniform).

```
PageRank

Input:
$P$, transition matrix of the random walk
$\alpha$, damping factor
$K$, number of iterations

Do:
$\pi \leftarrow \frac{1}{n}(1, \ldots, 1)$
For $t = 1, \ldots, K$, $\pi \leftarrow \alpha \pi P + (1 - \alpha)\frac{1}{n}(1, \ldots, 1)$

Output:
$\pi$, PageRank vector
```

The parameter α is known as the damping factor. Observe that the path length of the random walk until restart has a geometric distribution with parameter $1 - \alpha$. In particular, the average path length is:

$$\frac{\alpha}{1 - \alpha}.$$

Remark 3 *For the typical value $\alpha = 0.85$, the average path length is equal to 5.7, which is typical of the distance between two nodes of real graphs (cf. the six degrees of separation).*

The following result shows that the PageRank vector is the smoothing average of the distributions π_t of the pure random walk (without damping, $\alpha = 1$) at times $t = 0, 1, 2, \ldots$, with π_0 the uniform distribution.

Proposition 1 *We have:*

$$\pi^{(\alpha)} = (1-\alpha)\sum_{t=0}^{+\infty} \alpha^t \pi_t. \tag{6}$$

Proof. It is sufficient to check that the row vector $\pi^{(\alpha)}$ defined by (6) satisfies (5):

$$\begin{aligned} \alpha\pi^{(\alpha)}P + (1-\alpha)\frac{1^T}{n} &= \alpha(1-\alpha)\sum_{t=0}^{+\infty} \alpha^t \pi_t P + (1-\alpha)\pi_0, \\ &= (1-\alpha)\sum_{t=0}^{+\infty} \alpha^{t+1}\pi_{t+1} + (1-\alpha)\pi_0 = \pi^{(\alpha)}. \end{aligned}$$

□

Observe that $\pi^{(\alpha)} = \pi_0 + \alpha(\pi_1 - \pi_0) + o(\alpha)$. Since π_0 is the uniform distribution, the ranking is that induced by the sampling of a random neighbor when $\alpha \to 0$. When $\alpha \to 1$, the ranking is that induced by the limit of π_t when $t \to +\infty$ (proportional to the degrees if the graph is undirected).

The approximation provided by the first K jumps of the random walk (as in the algorithm described above) amounts to truncating the sum (6), namely to approximating $\pi^{(\alpha)}$ by

$$\alpha^K \pi_K + (1-\alpha)\sum_{t=0}^{K-1} \alpha^t \pi_t.$$

4 Personalized PageRank

While PageRank provides a global ranking of the nodes, it is interesting in practice to get a local ranking, relative to some target node(s). This is the objective of Personalized PageRank, used by Web search engines to display the most relevant pages relative to some request.

Let $s \in V$ be some target node. The idea of Personalized PageRank is to rank nodes relative to their frequency of visits for a random walk (re)starting from that node. Specifically, the transition matrix of the random walk is such that:

$$\forall j \neq s, \quad P_{ij}^{(\alpha)} = \begin{cases} \alpha\frac{A_{ij}}{d_i^+} & \text{if } d_i^+ > 0, \\ 0 & \text{otherwise.} \end{cases}$$

Observe that the random walk moves to node s with probability 1 from any sink, and with probability $1-\alpha$ from other nodes. The corresponding stationary distribution $\pi^{(\alpha)}$ is called the Personalized PageRank vector. The expression (6) remains valid, with $\pi_0 = 1_s^T$ (unit row vector on s) and π_t the distribution after t jumps from s (with restart to s from any sink). The parameter α controls the "locality" of the ranking, the successors of s being favored when $\alpha \to 0$.

The Personalized PageRank can be generalized to some set $S \subset V$ of target nodes, with relative weights captured by some distribution μ on S. The transition matrix of the random walk becomes:

$$\forall j \notin S, \quad P_{ij}^{(\alpha)} = \begin{cases} \alpha\frac{A_{ij}}{d_i^+} & \text{if } d_i^+ > 0, \\ 0 & \text{otherwise,} \end{cases} \quad \text{and} \quad \forall j \in S, \quad P_{ij}^{(\alpha)} = \begin{cases} (1-\alpha)\mu_j & \text{if } d_i^+ > 0, \\ \mu_j & \text{otherwise.} \end{cases}$$

The Personalized PageRank can also be computed by fixed-point iterations. Below is the pseudo-code in the particular case where the distribution μ is uniform on S.

```
Personalized PageRank

Input:
$P$, transition matrix of the random walk
$S$, set of target nodes
$\alpha$, damping factor
$K$, number of iterations

Do:
$\mu \leftarrow 1_S^T/|S|$
$\pi \leftarrow \mu$
For $t = 1, \ldots, K$, $\pi \leftarrow \alpha \pi P + (1-\alpha)\mu$

Output:
$\pi$, Personalized PageRank vector
```

It turns out that, in the absence of sinks, the Personalized PageRank vector associated with some distribution μ on S follows from the Personalized PageRank vectors associated with the nodes $s \in S$ taken individually. This is interesting from a mathematical point of view only; for computations, it is preferable to use the algorithm above, whose complexity is in $O(Km)$ independently of the cardinality of the set S.

Proposition 2 *In the absence of sinks, we have:*

$$\pi^{(\alpha)} = \sum_{s \in S} \mu_s \pi_s^{(\alpha)}, \tag{7}$$

where $\pi_s^{(\alpha)}$ is the Personalized PageRank vector associated with node s.

Proof. Observing that

$$P^{(\alpha)} = \alpha P + (1-\alpha) \sum_{s \in S} \mu_s 1 1_s^T,$$

with $P = D^{-1}A$, we get

$$\begin{aligned}
\pi^{(\alpha)}(\alpha P + (1-\alpha) \sum_{s \in S} \mu_s 1 1_s^T) &= \alpha \pi^{(\alpha)} P + (1-\alpha) \sum_{s \in S} \mu_s 1_s^T, \\
&= \alpha \sum_{s \in S} \mu_s \pi_s^{(\alpha)} P + (1-\alpha) \sum_{s \in S} \mu_s 1_s^T, \\
&= \sum_{s \in S} \mu_s (\alpha \pi_s^{(\alpha)} P + (1-\alpha) 1_s^T), \\
&= \sum_{s \in S} \mu_s \pi_s^{(\alpha)}.
\end{aligned}$$

□

Observe that the result is no longer valid in the presence of sinks, due to the restart mechanism.

5 BiPageRank

Now consider the case of a bipartite graph $G = (V_1, V_2, E)$ with $n = n_1 + n_2$ nodes, $n_1 = |V_1|$ and $n_2 = |V_2|$. The adjacency matrix A of a bipartite graph can be written:

$$A = \begin{bmatrix} 0 & B \\ B^T & 0 \end{bmatrix},$$

where B is the biadjacency matrix, of dimension $n_1 \times n_2$, specifying the edges between V_1 and V_2. Let $d_1 = B1$ and $d_2 = B^T 1$ be the vectors of degrees of nodes in V_1 and V_2, respectively. We denote by $D_1 = \text{diag}(d_1)$ and $D_2 = \text{diag}(d_2)$ the corresponding diagonal matrices. The vector of node degrees is:

$$d = A1 = \begin{bmatrix} d_1 \\ d_2 \end{bmatrix},$$

and the corresponding diagonal matrix is:

$$D = \text{diag}(d) = \begin{bmatrix} D_1 & 0 \\ 0 & D_2 \end{bmatrix}.$$

The random walk in this graph defines a Markov chain of period 2: starting from a node in V_1, the state of the Markov chain will be in V_1 in even times and in V_2 in odd times. The transition matrix is:

$$P = D^{-1}A = \begin{bmatrix} 0 & P_1 \\ P_2 & 0 \end{bmatrix},$$

where $P_1 = D_1^{-1}B$ is the transition matrix from V_1 to V_2 and $P_2 = D_2^{-1}B^T$ is the transition matrix from V_2 to V_1. In particular,

$$P^2 = \begin{bmatrix} P_1P_2 & 0 \\ 0 & P_2P_1 \end{bmatrix}.$$

Starting from V_1, the distribution of the Markov chain on even times has a limit π_1 that satisfies the balance equations:

$$\pi_1 = \pi_1 P_1 P_2. \tag{8}$$

Similary, starting from V_2, the stationary distribution on even times has a limit π_2 that satisfies the balance equations:

$$\pi_2 = \pi_2 P_2 P_1. \tag{9}$$

Proposition 3 *The stationary distributions π_1 and π_2 are proportional to d_1 and d_2, respectively.*

Proof. We have:

$$d_1^T P_1 P_2 = 1^T B P_2 = d_2^T P_2 = 1^T B^T = d_1^T,$$

which shows that $\pi_1 \propto d_1^T$. The proof is similar for π_2. □

We refer BiPageRank to the adaptation of PageRank to bipartite graphs. It is also based on a random walk with restarts. The only difference lies in the distribution used for restarts, taken uniform over V_1 or V_2, in the absence of personalization. We present the results of a uniform distribution over V_1. We refer to the BiPageRank vector as the Personalized PageRank vector $\pi^{(\alpha)} = (\pi_1^{(\alpha)}, \pi_2^{(\alpha)})$ with target set $S = V_1$. We have the analogue of Proposition 1, with $\pi(t) = (\pi_1(t), \pi_2(t))$ the distribution of the random walk after t steps of the pure random walk (without restarts) starting from the uniform distribution over V_1.

Proposition 4 *We have:*

$$\pi_1^{(\alpha)} = (1-\alpha) \sum_{t \in 2\mathbb{N}} \alpha^t \pi_1(t) \quad \text{and} \quad \pi_2^{(\alpha)} = (1-\alpha) \sum_{t \in 2\mathbb{N}+1} \alpha^t \pi_2(t). \tag{10}$$

Observe that when $\alpha \to 0$, the ranking of nodes in V_2 is that induced by the sampling of a random neighbor of V_1, while the ranking of nodes in V_2 is that induced by the sampling of a random neighbor of a random neighbor of V_1 (equivalently, this is the sampling of a random neighbor in the co-neighbor graph, as explained below). When $\alpha \to 1$, the rankings are proportional to the degrees, respectively the vectors d_1 and d_2.

```
BiPageRank

Input:
$P_1$, transition matrix from $V_1$ to $V_2$
$P_2$, transition matrix from $V_2$ to $V_1$
$\alpha$, damping factor
$K$, number of iterations

Do:
$\pi_1 \leftarrow \frac{1}{n_1}(1, \dots, 1)$
$\pi_2 \leftarrow 0$
For $t = 1, \dots, K$,
    $\pi_1 \leftarrow \alpha \pi_2 P_2 + (1 - \alpha)\frac{1}{n_1}(1, \dots, 1)$
    $\pi_2 \leftarrow \alpha \pi_1 P_1$

Output:
$\pi_1, \pi_2$, BiPageRank vectors
```

The personalized version of BiPageRank follows from some appropriate choice of the restart distribution, with support typically either in V_1 or V_2.

Co-neighbor graph. In view of (8) and Proposition 4, the BiPageRank vector $\pi_1^{(\alpha)}$ is equivalent to the PageRank vector in the co-neighbor graph $G_1 = (V_1, E_1)$ of adjacency matrix:

$$A_1 = BD_2^{-1}B^T,$$

with damping factor α^2. The interest of BiPageRank is that it does *not* require the computation and storage of this matrix. In practice, the biadjacency matrix B is sparse but the adjacency matrix A_1 may be dense, due to the small-world property (note that a node of degree k in V_2 generates a clique of k nodes in the co-neighbor graph G_1).

Directed graphs. BiPageRank can be applied to directed graphs as well. Consider a directed graph $G = (V, E)$ of n nodes with adjacency matrix A. This can be viewed as a bipartite graph of $2n$ nodes with biadjacency matrix A, each node being duplicated (one as a source of edges, the other as a destination of edges). The BiPageRank vector then corresponds to the distribution of a *forward-backward* random walk with restarts, where edges are followed in forward and backward directions alternately. The idea is similar to that used in the algorithms HITS [2] and SALSA [3].

To understand the difference with PageRank, consider some target node $s \in V$. Then the successors of s will be highly ranked using Personalized PageRank (at least for low values of α). In Wikipedia for instance, this means that the articles referenced in the target article will be highly ranked. Using Personalized BiPageRank, only nodes having many successors in common with s in G will be highly ranked. In the example of Wikipedia, it means that the articles having many common references with the target article will be highly ranked. The ranking is typically very different.

Undirected graphs. Similarly, BiPageRank can be applied to undirected graphs, viewing the adajcency matrix as a biadjacency matrix. It can then be interpreted as the ranking resulting from a 2-hop random walk, which is different from that induced by the 1-hop random walk of PageRank.

References

[1] Lawrence Page, Sergey Brin, Rajeev Motwani, and Terry Winograd. The PageRank citation ranking: Bringing order to the web. Technical report, Stanford InfoLab, 1999.

[2] Jon M Kleinberg. Authoritative sources in a hyperlinked environment. *Journal of the ACM (JACM)*, 46(5):604–632, 1999.

[3] Ronny Lempel and Shlomo Moran. SALSA: the stochastic approach for link-structure analysis. *ACM Transactions on Information Systems (TOIS)*, 19(2):131–160, 2001.