CS838-1 Advanced NLP: Link Analysis

Xiaojin Zhu

2007

Send comments to jerryzhu@cs.wisc.edu

Many NLP problems can be formulated as graphs. For example, Web pages form a directed graph with hyperlinks. What does the graph structure tell us about the importance of each Web page?

1 Hubs and Authorities

We are interested in two kinds of Web pages:

1. authority: a Web page with good, authoritative content on a specific topic;
2. hub: a Web page pointing to many authoritative Web pages.

In fact, we will change the definition of authoritative Web pages a bit, since we only see the graph structure, not the actual page content.

1. authority: a Web page that is *pointed to* by many hub pages.

This is motivated by the fact that if an authoritative Web page has good content, then many "yellow page"-ish hub pages will point to it. The definitions are circular. Interestingly well-defined hubs and authorities can be obtained as follows.

Let there be n Web pages. Define the $n \times n$ *ajacency matrix* A such that

$$A_{uv} = \begin{cases} 1 & \text{if there is a link from } u \text{ to } v \\ 0 & \text{otherwise.} \end{cases} \tag{1}$$

Each Web page has an authority score a_i and a hub score h_i. We define the authority score of the i-th Web page by summing up the hub scores that points to it, normalized:

$$a_i \propto \sum_{j=1}^{n} h_j A_{ji}.$$

This can be written concisely with matrix operation on the vector a:

$$a \propto A^\top h.$$

Similarly we define the hub score to be the sum of authority score of the Web pages it points to:

$$h_i \propto \sum_{j=1}^{n} a_j A_{ij},$$

or

$$h \propto Aa.$$

Let's start arbitrarily from $a_0 = 1, h_0 = 1$, where 1 is the all-one vector. Repeating these and we find that

$$\begin{aligned} a &\propto (A^\top A)^\infty A^\top 1 \\ h &\propto (AA^\top)^\infty 1. \end{aligned} \tag{2}$$

Recall x is an eigenvector and λ the corresponding eigenvalue of matrix M, if

$$Mx = \lambda x.$$

x is called the leading eigenvector, if λ is the largest eigenvalue among all eigenvalues of M. If there is one unique largest eigenvalue, then only the leading eigenvector (up to scaling) will survive the iterations in (2). This is known as the power method.

In the end, a is the leading eigenvector of $A^\top A$, and h is the leading eigenvector of $AA^\top$. The hubs and authorities can be found by thresholding h and a, respectively.

2 PageRank

This algorithm is proposed by Google and assigns a single importance to each Web page. Think of a *random walk* on the graph. Let the random walker be at node u, which points to N_u nodes. The walker randomly picks one of the N_u nodes to walk to, with probability $1/N_u$. The process repeats.

Let P be the $n \times n$ *transition matrix*, such that

$$P_{uv} = P(v \to u) = \begin{cases} \frac{1}{N_v} & \text{if there is a link from } v \text{ to } u \\ 0 & \text{otherwise.} \end{cases} \tag{3}$$

Note the direction. P is column-normalized:

$$\sum_u P_{uv} = 1.$$

Let r_0 be the probability vector of the initial position of the walker. The position of the walker after one step is described by

$$r_{1u} = \sum_v r_{0v} P_{uv},$$

or compactly

$$r_1 = Pr_0. \tag{4}$$

The random walk is meant to model the behavior of a 'random Web surfer' who aimlessly follows hyperlinks. However the surfer could be trapped by a self pointer; There could be disconnected components of the Web that, depending on the initial position, are never reached. Therefore we introduce a *teleporting* scheme: at each step, the walker flips a coin. With large probability α, the walker will perform the above random walk via an edge. With small probability $1-\alpha$, however, the walker is teleported to a random node v with probability b_v. We thus have

$$\begin{aligned} r_{t+1} &= \alpha P r_t + (1-\alpha) b && (5) \\ &= (\alpha P + (1-\alpha) b 1^\top) r_t. && (6) \end{aligned}$$

Let $M = \alpha P + (1-\alpha)b1^\top$ and we see this is again the power iteration $r \leftarrow Mr$. Therefore r is the leading eigenvector of M.