Text and web mining – part II

Wholly new forms of encyclopedias will appear, ready made with a mesh of associative trails running through them, ready to be dropped into the memex and there amplified. (Vannevar Bush, 1945)
Using hyperlinks to rank web pages

• Problem: given a query, how to retrieve a set of high quality and relevant pages from the Web.

• In scientific communities, a paper is considered of good quality if it is cited by other good quality papers (citation analysis in Bibliometrics)

• Analogy: a candidate for employment is valued if many other valued people are prepared to recommend him
Prestige in social networks: recommendations from (or relationship with) high-rank individuals (above) are more effective to reach a high rank than recommendations by low-rank ones (below).
Using hyperlinks to rank web pages (3)

• After a seminal paper by Marchiori about the importance of hyper-information (information in the hyperlinks), Larry Page and Sergey Brin developed the PageRank algorithm

• Same social networks principles, by substituting “recommendations” and “citations” with hyperlinks
Using hyperlinks to rank web pages (4)

• The **prestige** of a page is related to how many pages of **prestige** link to it.

• Note the **recursive definition**: to calculate the prestige, one needs to start from prestige values of other pages, and so on...

• Start with initial prestige values (random?) and **iterate**, if prestige values **converge** the problem is solved.
Using hyperlinks to rank web pages (5)

- What guarantee that the process converges, hopefully to the same limiting distribution, not depending on the initial distribution of values?

- The solution to this problem is related to basic linear algebra concepts of eigenvalues and eigenvectors, as well as Markov chains.
Using hyperlinks to rank web pages (6)

• Let’s use **linear definitions**

• To calculate rank of a page:
  – Examine **incoming links** (the hyperlinks of other pages pointing to the given page).
  – Each incoming link from page i contributes an **addendum** equal to the rank of i divided by the number of i’s outgoing links
Using hyperlinks to rank web pages (7)

Recalculating the rank of a page in PageRank. The initial rank is distributed along the outgoing links (adapted from the original paper).
Using hyperlinks to rank web pages (8)

• New rank values $p_k$ at iteration $k$ are obtained by a linear transformation of the previous values through a matrix $M$ (derived by analyzing hyperlinks)

$$p^k = M p^{k-1}$$

• After $k$ recalculations:

$$p^k = M^k p^0$$
Using hyperlinks to rank web pages (9)

• Assume a **basis of eigenvectors** of $M$ exists

• let $\lambda_1, \lambda_2, \ldots, \lambda_n$ be the $n$ **eigenvalues**, and $v_1; v_2; \ldots; v_n$ the corresponding **eigenvectors**

\[ M v_i = \lambda_i v_i \]

• $\lambda_1$ is the **dominant eigenvalue**, so that $|\lambda_1| > |\lambda_j|$ for $j > 1$.

• The initial vector $p_0$ can be written as a linear combination of the basis vectors:

\[ p^0 = c_1 v_1 + c_2 v_2 + \cdots + c_n v_n. \]
Using hyperlinks to rank web pages (10)

- By linearity and definition of eigenvectors:

\[
M^k p^0 = c_1 M^k v_1 + c_2 M^k v_2 + \cdots + c_n M^k v_n \\
= c_1 \lambda_1^k v_1 + c_2 \lambda_2^k v_2 + \cdots + c_n \lambda_n^k v_n \\
= c_1 \lambda_1^k \left( v_1 + \frac{c_2}{c_1} \left( \frac{\lambda_2}{\lambda_1} \right)^k v_2 + \cdots + \frac{c_n}{c_1} \left( \frac{\lambda_n}{\lambda_1} \right)^k v_n \right)
\]

- All terms tend to zero apart from the one proportional to the dominant eigenvector.

- A simple iteration of matrix multiplication, after starting from almost arbitrary initial conditions extracts the dominant eigenvector!

- **Power method** for obtaining the dominant eigenvector by the power of a matrix: \(M^k\)
A surprising connection with web surfing

• Modeling the movement of a **web surfer** on the various web pages

• Let $E$ be the adjacency matrix of the web:

$$ (u, v) \in E \quad \text{(or} \quad E_{uv} = 1 \text{)}$$

if and only if there is a link from page $u$ to page $v$.

• What is the probability $p_v^1$ of the surfer being at page $v$ after one step?

• Let $N_u = \sum_v E_{uv}$ be the out-degree of page $u$

• After defining

$$ L_{uv} = \frac{E_{uv}}{N_u} $$
A surprising connection with web surfing (2)

• One obtains:

\[ p^1_v = \sum_u L_{uv} p^0_u \quad \text{or} \quad p^1 = L^T p^0 \]

• After \( i \) steps:

\[ p^i = L^T p^{i-1} \] (again, iteration of matrix multiplication)

• If \( E \) is irreducible and aperiodic, \( p \) converges to the largest eigenvector

\[ \lim_{i \to \infty} p^i = p \]

• the prestige (rank) of a page can be interpreted also as **probability that a random surfer following links will be found at a given page**
A surprising connection with web surfing (3)

- Real-world transition matrices: Web is not strongly connected, and that random walks can be trapped into cycles.

- “damping factor” corresponding to a user with an arbitrary probability $d$ of going to a random page (even unconnected) at every step.

- The transition becomes ($\mathbf{1}$ is the identity matrix)

$$p^i = \left( (1 - d) L^T + \frac{d}{N} \mathbf{1}_N \right) p^{i-1}.$$
A surprising connection with web surfing (4)

• The eigenvector corresponding to the largest eigenvalue can be obtained:

  - Start with random vector \( p \leftarrow p^0; \)
  - repeat:
    - update vector:
      \[
      p \leftarrow \left( (1 - d)L^T + \frac{d}{N} 1_N \right) p;
      \]
    - from time to time, normalize it:
      \[
      p \leftarrow \frac{p}{\|p\|_1}.
      \]

Normalization to avoid very large components and numerical problems. One is not interested in absolute prestige values but in relative ones.
Identifying hubs and authorities: HITS

• A different analysis of the web. In a scientific community good articles are either **seminal** (i.e., many others reference to them) or **surveys** (i.e., they reference to many others).

• In the web, pages may be **authorities** or **hubs**. For example portals are very good hubs.

• Two score measures, called hubness and authority:

\[ h = (h_u), \quad a = (a_u) \]
Identifying hubs and authorities: HITS (2)

- HITS algorithm (Hyperlink-Induced Topic Search)
- Given query \( q \), let \( R_q \) be the root set returned by an IR system. The computation is performed only on this result set.
- The expanded set is formed by adding all nodes linked to the root set:

\[
V_q = R_q \cup \{u : ((u \rightarrow v) \lor (v \rightarrow u)) \land v \in R_q\}.
\]

- Let \( E_q \) be the induced link subset, \( G_q = (V_q, E_q) \).
Identifying hubs and authorities: HITS (3)

- Authority and hub values are defined in terms of one another
- **hub score** $h_u$ proportional to sum of referred authorities, **authority score** $a_u$ proportional to the sum of referring hubs

\[
\begin{align*}
    a &= E^T h \\
    h &= E a.
\end{align*}
\]
Identifying hubs and authorities: HITS (4)

• The iterated method in HITS:
  - initialize $a$ and $h$ (e.g., uniformly);
  - repeat:
    - $h \leftarrow Ea$;
    - $a \leftarrow E^T h$;
    - normalize $h$ and $a$.

• The top-ranking authorities and hubs are reported to the user.

• The principal eigenvector identifies the largest dense **bipartite subgraph**.
Clustering in web mining

- During web search, to avoid overloading the user, identify groups of closely related documents, show only a small number of representative prototypes.
- Queries can be ambiguous. For example, star: movie stars, or celestial objects?
- Mutual similarities in term vector space can help clustering.
Clustering in web mining (2)

• Retrieved pages can be characterized either **internally** by some intrinsic property (e.g., terms contained, coordinates in TF-IDF space) or **externally** by a measure of distance between pairs. Examples are: Euclidean distance, dot product, Jaccard coefficient.

• After defining the metric, the usual **bottom-up** or **top-down clustering** techniques can be used (see chapter on clustering).
Gist

• The Web is a vast expanse of data, some of it structured, some partially structured or not at all.

• **Crawling and indexing** are systematic methods to visit web pages, harvest the information contained therein and **prepare data structures for searching**, information retrieval and **ranking**.

• By **transforming text into vectors of data** (e.g., frequencies of selected words as in the vector-space model) some traditional ML techniques can be reused, but the **richer amount of structure in web documents** permits a more focused analysis.
Gist (2)

- Web-mining find explicit relationships between documents (web links), infer implicit ones (by clustering), rank the most relevant pages or identify the most relevant and well-connected persons in a network of people.

- Abstraction helps to use similar tools for networks of pages and networks of people. As a notable example, the use of hyperlinks and linear algebra tools (eigenvectors and eigenvalues), previously used to rank researchers in bibliometrics, leads to a very powerful technique to rank web pages, now at the basis of Google search-engine technology.