T-122.101 Modeling and Mining the Web

Link Analysis 2

Anne Patrikainen
Laboratory of Computer and Information Science
22.10.2003

Outline of the Presentation

- Last week we introduced two famous link analysis algorithms, PageRank and HITS
- Are these algorithms stable under small perturbations of the Web graph?
- Better algorithms: SALSA and Probabilistic HITS
- How to fool search engines?
- How good are the rankings of PageRank and HITS when compared to human experts?

Stability of the link analysis algorithms

- Nobody is able to crawl the whole Web
- Moreover, the Web is changing all the time
- How do modest changes in the Web graph affect the results of link analysis algorithms?
- Consider adding and removing a small percentage of the edges and vertices of a graph and running PageRank and HITS before and after
- We would hope that the rankings remained relatively close to the original values

More on stability

- Experimental evidence suggests that HITS is not stable under small perturbations of the graph, but PageRank performs better
- Ng. et al. (2001) have shown that the stability of HITS depends on so-called eigengap, the difference between the two largest eigenvalues of $A^T A$ (same for AA^T)
- They have also shown that for PageRank, stability is better if the teleportation probability ϵ is larger
- Furthermore, perturbations in pages of small rank have small effect

SALSA 1/3

- PageRank is more stable than HITS but HITS has a nice two-level ranking scheme
- Why not combine these two?
- SALSA (Stochastic Approach for Link Structure Analysis) does this. It was introduced by Lempel and Moran (2001)
- There are also a few other approaches, for instance Probabilistic HITS, but they are very similar

SALSA 2/3

- The idea is to follow hyperlinks both in the forward and in the backward directions
- We start with our graph and construct a bipartite graph consisting of a set of hub side vertices and a set of authority side vertices
- Two separate random walks are introduced
- One random walk starts from the hub side, and in one step follows a link to the authority side and some other link back to the hub side
- The other random walk starts from the authority side

SALSA 3/3

- The hub side random walk has a transition probability matrix $\tilde{H}$. The hub scores are calculated as the stationary distribution of the corresponding random walk (i.e. the principal eigenvector of $\tilde{H^T}$)
- $\tilde{H}_{ij} = \sum_{i_h \to k_a, k_a \to j_h} \frac{1}{\deg(k_a)\deg(j_h)}$
- The authority side random walk has a transition probability matrix $\tilde{A}$. The authority scores are similarly calculated as the corresponding stationary distribution
- It has been shown theoretically that SALSA is more stable than HITS in some situations

Probabilistic HITS, 1/3

- Probabilistic HITS is also known as Subspace HITS
- Recall that the HITS authority scores are calculated using the principal eigenvector of $A^T A$, where A is the adjacency matrix of the graph
- However, the rest of the eigenvectors might contain relevant information as well
- Moreover, subspaces spanned by the first few eigenvectors are more stable than the eigenvectors themselves under perturbations of the graph
- The idea is to take several eigenvectors into account

Probabilistic HITS, 2/3

- We will calculate the k first eigenvectors of $A^T A$ and AA^T
- There are no sound theoretical results on how to combine the multiple eigenvectors
- One idea is to calculate the rank of the i:th page by taking the i:th element of each of the k eigenvectors, weigh the elements by the corresponding eigenvalues, and take the sum
- Several other weighing schemes have been experimented with
- In any case, the resulting algorithm seems to be more stable than original HITS

Probabilistic HITS, 3/3

- Probabilistic HITS can be viewed as LSI (Latent Semantic Indexing) for the co-citation and bibliographic coupling matrices!
- AA^T is the co-citation matrix, in which the entry (i, j) represents the number of pages cited (or linked) by both pages i and j
- A^TA is the bibliographic coupling matrix, in which the entry (i, j) represents the number of pages that are citing (or linking to) both pages i and j
- In traditional LSI, we consider a space of terms; in case of the co-citation matrix, we consider a space of citations

Search Engine Persuasion 1/2

- Search Engine Persuasion (SEP) refers to websites faking high relevance with respect to some topics for the sole purpose of attracting visitors
- In first-generation search engines, page ranking was based on the text content of the pages
- SEP was simple: It was enough to use a lot of suitable keywords in the page source, often invisible to the human surfer
- Second-generation search engines employ link analysis algorithms, and are praised for making SEP difficult

Search Engine Persuasion 2/2

- SEP is possible also with second-generation search engines
- A *link farm* is a densely connected Web subgraph artificially built for the purpose of gaining high PageRank of HITS scores
- Consider a group of collaborating n websites
- In addition to the actual content, each site has a hub page pointing to the rest $n-1$ sites
- Now each site has $n-1$ in-links from hub pages having $n-1$ out-links to sites having $n-1$ in-links...
- Link farms can be sometimes detected by looking at Web topology, but other clever SEP schemes are constantly being developed

Empirical study on page ranking

- How do HITS and PageRank perform when compared to human expert rankings?
- Amento et al. (2000) have reported an empirical study in which 16 human experts ranked web pages in five topics related to TV entertainment and music
- The study revealed that ranking documents according to HITS authority score or PageRank score yields high precision for the top 5 or 10 ranked documents
- However, simpler ranking schemes such as page indegree or total number of pages in the website performed equally well

Homework problems for Link Analysis 2

In this homework assignment, you will conduct a small stability experiment. Choose a graph data set, for instance the sample of the Web you have crawled. Implement traditional HITS and SALSA. Run these first for the whole data set and store the vertex rankings. Then, remove 10% of the edges and vertices, re-run the algorithms, and store the rankings. Repeat for 3 times. Compare the results of the repeated experiment for each algorithm. Are the rankings preserved? Why, why not?

By conducting the experiment for HITS you will receive one exercise point. By repeating for SALSA you will receive another point.