Data Mining Assignment IV
Web Searching using the HITS algorithm

Submit your solutions to: Per.Gustafsson@it.uu.se

Due date: 21/12/2005, 17:00

1 Background

The purpose of this programming assignment is to implement the HITS algorithm [1] to calculate the Hub and Authority value of a set of web pages related to a certain subject. This information can be used to decide which pages in the set can be considered the greatest authorities on the subject and which pages can be considered best at linking to good pages within the given subject area.

2 Assignment Description

Assume that an input file named links.txt consists of ASCII text that looks as follows:

```
http://www.it.uu.se/index.html http://www.uu.se http://www.kth.se/
http://www.kth.se www.uu.se http://www.slu.se
http://www.uu.se http://www.it.uu.se www.dn.se
http://user.it.uu.se/~pergu http://www.uu.se http://www.it.uu.se/
http://www.slu.se
```

Each row can be interpreted in the following fashion:

> The first web page in each row contains links to the rest of the web pages in that row. The information is not complete: some web pages that are linked to by another webpage do not appear at the beginning of a line. Some rows possibly only contain one web address (like the last one in the above example) signifying that that web page does not link to any other web page.

Note that some links appear in different forms e.g.

- http://www.it.uu.se/index.html
- http://www.it.uu.se/
- http://www.it.uu.se

All are linking to the same page despite the fact that they all look different. Another example is:

- www.uu.se
- http://www.uu.se

Where one of the links is preceeded by 'http//' and the other is not. They are still linking to the same page though. Since the textfile can contain these different types of links you need to preprocess the file so that all links that point to the same page have the same name.

Your task is to write a program (in your favourite programming language) which calculates the hub weight and the authority weight of each web page in the file. The only parameter your program need to take is the filename. (If you use the iterative algorithm to calculate the weights you can also let the number of iterations be an input parameter). The output of the program should be the 10 most authoritative pages in the collection together with their respective authority weight as well as the 10 most "hubby" pages together with their respective hub weights; see Figure 1 which shows an example.

Your program needs to first preprocess the files so as to recognize the link names which are linking to the same page and remove all the links for which there is no information about what they are linking to.

```
> myHITS links.txt

Pages with high authority
==========================================
Address                   Authority value
==========================================

http://www.uu.se              0.8363
http://www.it.uu.se           0.3954
http://www.kth.se             0.2685
http://www.slu.se             0.2685
http://user.it.uu.se/~pergu   0.0000

Pages with high hubbiness
==========================================
Address                      Hub value
==========================================
http://user.it.uu.se/~pergu   0.6072
http://www.it.uu.se           0.5446
http://www.kth.se             0.5446
http://www.uu.se              0.1949
http://www.slu.se             0.0000
```

Figure 1: Example use of the program.

The only thing you need to check for in order to find different links pointing to the same page are trailing '/', trailing '`/index.html`' and preceeding '`http://`'.

An example of a possible session using your program on the data of file `links.txt` above is given in Figure 1.

To test your implementation you will be given four different files. Each file contains web pages from a certain topic (They appear in the assignments' page and are called `links_*.txt`). The files have been created using the method described in [1]. Analyze the results of your program to find out whether or not the pages with high hub or authority value truly are "hubby"/authoritative with respect to the topic.

3 Report

As in all assignments of this course, you are allowed and encouraged to work in pairs. Your report should contain information about:

- Your names, e-mails, and course to which you are registered.
- Pointer to a set of source files of your implementation. As in previous assignments, you are not allowed to update these files after the deadline. Since the implementation language is unspecified, your report should also contain a *clear and detailed* explanation of how to (make and) run your program.
- A discussion about the results that you got running the program on the supplied files.
- A discussion about how you chose to preprocess the data and whether there are other techniques that could be used to test if two URLs point to the same page

References

[1] David Gibson, Jon Kleinberg, and Prabhakar Raghavan. Inferring Web Communities from Link Topology. In *Proceedings of the Ninth ACM Conference on Hypertext and Hypermedia: Links, Objects, Time and Space - Structure in Hypermedia Systems*, pages 225-234, June 1998.