

Project description

Project 1 – Web structure Mining Automatic Web chart

Web mining, HIA, 2004

The web crawlers

Thomas Andersen
Quang Van Nguyen
Trond Undrum



1. Project description:

1.1 The assignment

Our task is to create a web crawler. The robot will be given a webpage where it is supposed to find all links and find the relationship between the linked pages.

1.2 Our solution

The robot will be given a webpage from a user and it will download the source code. The source code which is downloaded is the source code visual to a browser (regular HTML and java-scripts and so on). All server interpreted code will be ignored by the program.

When the source code is downloaded all links are found and put in a table. A link is defined as the value of the href-attribute in an a-tag ()

All the links will be classified. A link can be classified as an "internal" link, "external" link, or a "not page link".

An internal link is a link to a page in a deeper level of the given page.

Example: www.hia.no is the given page, www.hia.no/student is an internal link found on the page.

An external link is a link to a page which is not in a deeper level of the given page.

Example: www.hia.no is the given page, www.vg.no is an external link found on the page.

A not page link is a link which not leads to another webpage, but to an other file or a mail address.

Example: www.hia.no is the given page, mailto:gunnar@hia.no, or www.hia.no/sounds/sound.wav is a link found on the page.

The program will be a CGI-program that can be started from a webpage and not from the users computer.

All the internal links will be collected in a new list.

The list with the internal links will proceeded and treated as new pages. All links inn these pages will be followed to the pages they are pointing to, and all links on these pages will be analyzed as the first page.

When the analyzing of all links are finished a representation of the result will be given.

The representation is supposed to be given as a graphical image of a graph with directions. But we will make this in two steps. The first step would be to create a figure just with text to illustrate the result of the work.

Index page

Internal link 1

Internal link 1.1

Internal link 1.2

Internal link 1.3

Internal link 2

Internal link 2.1

Internal link 2.2

Internal link 2.3

Internal link 3

Internal link 3.1

Internal link 3.2

Internal link 3.3

Fig. 1.2.A: Text, illustrating the result of the work.

When we know that the algorithm work we will create a graph with directions.

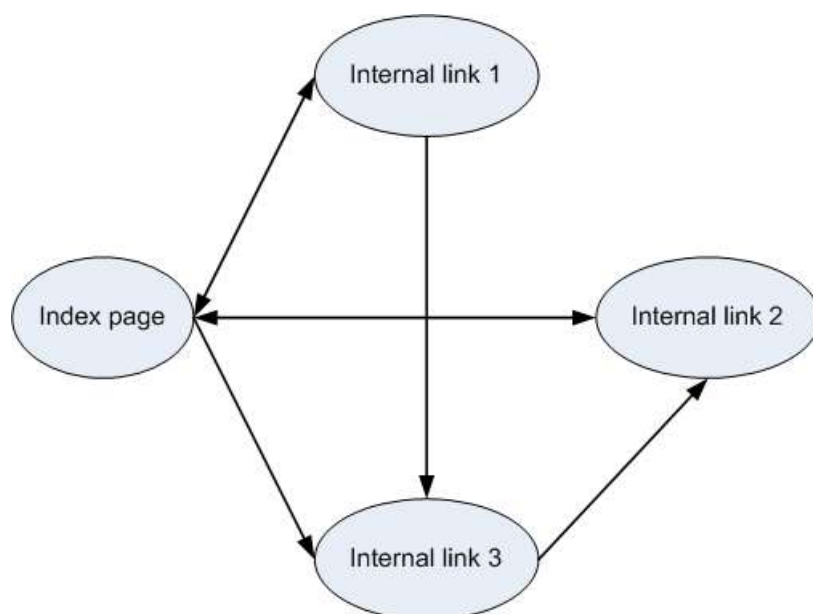


Fig 1.2.B: Graph with directions

Both of these presentations will be given on the same webpage the user uses to start the program.

1.3 Work strategy and plan

Programming	Date to be finished
Writing the start of the program	DONE
Write the code which finds the webpage and downloads source code	DONE
Write the code which finds and collects all hyperlinks	DONE
Decide for and find an algorithm for classification	DONE
Create or find training examples with html	DONE
Program the algorithm for classification	DONE
Test the algorithm hard	31-1-2004
Expand the algorithm to create a directed graph	15-10-2004
Write CGI support for the program and test the program from a webpage. The webpage is a strict xhtml 1.1 webpage.	18-10-2004
Create algorithm and program the part which analyzes	8-11-2004
Create the graphical user interface	10-11-2004

Other tasks	Date to be finished
Write the project description	30-09-2004
Write the project report	18-11-2004
Project presentation	25-11-2004

1.3 Content of the project report

1. Abstract
2. Table of content
3. Introduction
4. Problem description
5. Solution
6. Functionality
7. Test plan and results
8. Further development
9. Conclusion
10. References

This list of content could be changed later.