Appendix A: Title and short description of the research project:

The Analysis of possibility patterns appearance in url

Preface:

Thesis title:
Communication among information systems

Author:
Ing. Roman Hujer

Supervisor:
RNDr. Tomáš Vaníček Ph.D.

Department:
Department of Applied Informatics,
Faculty of Civil Engineering,
Czech Technical University in Prague

Abstract:

The Internet contains a lot of information. This information is being organized into resources and each resource is identified by an url. I am focusing on url from a semantic point of view. Passing trough the network is traditionally based on a hyperlink concept, where you can move from one resource to another only if they are interconnected with a hyperlink. If a pattern could be found in url, it could serve as a base of a new way for navigation trough the network.
Although the topic of my work is close to SEO, I do not intend to go deep into SEO problematics, as far as it is be possible.

Motivation:

In the Internet there are a lot of resources, which cannot be accessed from another resource via hyperlink. You can retrieve them only when you know exact url. Also there are resources, which are accessible in standard ways, but they could be also accessible in different ways. And it could be beneficial to access them in another way.

Research questions:

- Could analyzing semantics of url lead to previously unrevealed resources?
  - What is the effectiveness of searching new resources (targets) based on analyzing url?
  - Is there any advantage against traditional approach of following hyperlinks, even though a target could be accessed in both ways?
  - What is the impact of this technique to pageranking?
- Could be identified common behavior in process of creating url?

Hypothesis draft:

- Analysis of an url can reveal additional resources, but it is balanced out by the decreased effectiveness.
- Pattern in url could be a base for traversing among resources, which are not interconnected with a hyperlink.
- There could be established a hyperlink classification.

Possible Audience:

- search engines
- designers of web-based social networks
- companies involved in web advertising

Used methods

I will use empiric, statistic and inductive methods.
Primary method used in my work is empiric. It is crawling over the network. I will write one or more programs, which (together) will allow me to crawl over the network and collect data.
Then I will analyze the data and I will use statistic methods to evaluate it. Results of data analysis will allow me to fine-tune the crawler or likely be an input to crawler again.
After collecting enough data I will use inductive methods to analyze discovered patterns and try to establish url classification.

Current achievements:

- I developed an experimental web-crawler called "Sumid", with the following capabilities:
- Analyzing based on a number in url. I consider url as a path in tree. Analysis currently applies only to leaf elements. (Which is just a technical issue.) The number is going to be considered as one example of a regular expression. One of the goals is to find other regular expressions, which can reveal semantics of url.
- The urls for analyzing are provided manually as a text "linklist" file (xml form of linklist discouraged by the supervisor). I assume that there could be another component which will provide url suitable for the analysis. Linklist could be a common interface between Sumid and an external url provider.
- Action performed if target found is generally called "operation". Operation currently wired in Sumid is simple download. It is a kind of "dummy action" instead of something more sophisticated, like for example indexation. Currently this dummy action is good enough. (In other words: change from download to indexation has low priority.) Download and indexation are special cases of general "operation".
- Currently successfully reached target's url are not recorded. (Which is just a technical issue.) This record will be used as a base material to figure out how urls tend to be created.

Possible issues:

The first issue I expect is an ethical problem. Thus it is completely out of scope of my work, although I fell I should mention it. People got used to how search engines work and they are used to certain way of navigating trough the Internet. If they want something to be visible, they have to point a link to it. If they did not set a hyperlink which points to their resource, they consider it invisible. But that is not true. They should have to restrict access to the resource, otherwise it is public. So, is it ethically acceptable to reveal these "invisible" resources? Of course some solutions for this issue have already been developed. For example well known file robots.txt. But I rather prefer the idea that everything which is published on the Internet should be secured, or considered as public. But this will be always disputable.

The important aspect is an impact on pageranking. Technology of pageranking is currently implemented in most (if not all) todays search engines. One of the criteria used in pageranking is “how many hyperlinks point to a resource”. If I pass to the resource without using hyperlink, I cannot count it. The simplest answer is that the ranking gain from the incoming links will be zero. Would not be the decrease in relevance too high? My current assumption is that it is better to have a resource with a small relevance than not having it at all.

The internet is too big to cover and it is growing every day. I could perform the research only on a small subset of it. There is a risk that the disability of finding patterns in url is only because of an incorrectly selected subset. Also major patterns in my subset could been marginal on the Internet and vice versa.

The worst possible issue reinventing the wheel. It is possible that a similar research has already been performed. It could be publicly available and I may have just overlooked it. Or it could be proprietary and I cannot know about it.

Relevant Literature:

Below I state scientific literature which supports relevance of my research questions and proposed approach and will be cited in my thesis.

Sen P, Namata G, Bilgic M, Getoor L, Gallagher B , Eliassi-Rad T
Collective Classification in Network Data
AMER ASSOC ARTIFICIAL INTELL (2008)

Cun-He, Li; Ke-qiang, Lv
Hyperlink classification: A new approach to improve PageRank
IEEE COMPUTER SOC (2007)

Kamvar S, Haveliwala T, Golub G
Adaptive methods for the computation of PageRank
ELSEVIER SCIENCE INC (2004)

Haveliwala TH
Topic-sensitive PageRank: A context-sensitive ranking algorithm for Web search
IEEE COMPUTER SOC (2003)

Getoor L
Link mining: a new data mining challenge
ACM (2003)

Yoshiaki Mizuuchi, Keishi Tajima
Finding context paths for Web pages
ACM (1999)

Academic stay gains:

At the University of Trieste I would like to enrich my point of view to web technologies and compare ideas with experts in Trieste. Under the supervision of Prof. Alberto Bartoli I would like to gain deeper knowledge of web security, classification and data mining on web. I would like to participate in a research project carried out by Computer Networks Programming Lab.

I also want to gain an international experience. I want get to know the environment of University of Trieste and learn how the research is conducted here.

Appendix B: Brief explanation of terms used in the area of the problem

Sumid: It was an acronym for: Script used for mass items downloading. While focus of the script was shifted from downloading to url analyzing, the name remained the same.

Resource: Any object identified with unique url/uri accessible through Internet.

Traversing: The actual step from one resource to another, when the resources are not connected by hyperlinks.

Target: Resulting url (or a resource located on this url) after performing traversal.

The idea of url as a tree: Root node of the tree is always "http" (so far I am not going to analyze other protocols). Then every slash is an edge. Every string between two slashes is a content of a node. Then url is always a path from the root to a leaf. If I change any letter in the content of a node, I get the content of a different node. Also I get a path from the same root to a different leaf. If I allow to change the length of the content of a node (which is reasonable), I get an infinite tree. That means the tree with an infinite number of nodes, but with a constant length of the path from the root to a leaf. There is a finite subtree of urls, which actually do exist.

Crawler: or robot, bot, spider ... etc are different names for the same thing. An artificial program/script parsing content of resources on the Internet, which was primarily designed for humans, in order to perform an “operation”. The operation might be to extract some data and usually find a way to another resource.

Following of hyperlinks: A traditional way of moving from a web page (or another resource) to other is following of hyperlinks. This is the base idea of hypertext. Also todays search engines rely on hyperlinks during the indexing process. But this requires an existing hyperlink. If a hyperlink does not exist or it is faulty, pass to the target resource is not possible.
What shall I do if the hyperlink does not exist? Or what if two resources are interconnected indirectly? Then I could use a different way for passing from one resource to another.
If a hyperlink is faulty then I can find that it does not match the pattern, so I can correct it with the use of the pattern.

Linklist: A list of links. A linklist file is an input for Sumid and it can be a part of an output as well. It is a text file where each line contains an url. From the point of view of a program operation, each line is a seed. In the linklist it is also allowed an empty line and a comment. The comment starts with # (hash).
I was also considering xml form of the linklist. XML linklist could carry additional parameters for each url, which would make Sumid operation much more effective. The same format of xml linklist could have been used for Sumid outputs. But editing xml as an input will make the control of Sumid more complicated. The possibility of passing input parameters in the linklist will make the user tend to make many manual edits to the linklist, which is undesirable in the large scale.