Assistance of Web browsing by indicating the future Web pages

NAGINO Norikatsu YAMADA Seiji
CISS, IGSSE, Tokyo Institute of Technology
{nagino,yamada}@ymd.dis.titech.ac.jp

Abstract

This paper describes a novel method to assist a user in gathering interested Web pages on the WWW by narrowing a search space using sequences of the user's current browsed pages. Various methods of gathering interested Web pages for a user on the WWW have been proposed. Those approaches can't make an agreeable response to user's interest shift dynamically. We think those approaches don't try to understand which links will be selected and what kind of pages will be interested by a user in the past and the future.

To cope with this problem, rules to select a link which a user wants see are introduced in our system, and the rules called "**Navigation Rule**". Each rule has a class of Web page types in the condition part and a link-type in the action part. Also they are weighted for indicating the preference. Every time user selects a link to see the next page, the weights of matched rules are increased. The rule having the highest weight value was applied and some matched Web pages are fetched by Web robots again and again. We called the search strategy "**Navigation Search**". Web robots don't gather some useless Web pages and the search space will be narrow using the **Navigation Search**. Also we describe the way to provide an interface which display Web pages gathered by Web robots. A user can understand that a user will be reach what kind of Web pages with this interface. Also using the interface, a user also be able to select some links to interest Web pages directly without crawl all path.

1 Introduction

The accessible information through the Internet is increasing explosively as the WWW becomes widespread. In this situation, the WWW is very useful for a user who wants to gather interesting information. However there is a significant issue that a user does not know where the information exists. A practical and simple solution of the problem is to use a search engine like MetaCrawler, AltaVista, YaHoo with the interesting information as a query. The search engine provides a list of relevant Web pages to a user. Unfortunately, since a database of a search engine is very huge and adequate filtering is hard, many Web pages including irrelevant ones may be indicated.

To cope with this problem, various methods of gathering interested Web pages for a user using Web robots on the WWW have been proposed. WebWatcher[2] and Letizia[4] are able to indicate the Web pages that a user wants to see next. Using browsing history, they learn to predict useful Web pages for a user. However these systems do not consider sequences of browsed Web pages. FishSearch[1] and InfoSpiders[6] are distributed online search algorithms based on technique of artificial life for gathering relevance information. These approaches can't make an agreeable response sufficiently to user's browsing shift dynamically. PWM[8] is search algorithm for gathering interest Web pages with user's anytime controlling Web robots. In this system, gathered Web pages was divided into clusters, and user selects can select clusters about which he/her wants know more. Web robots refer the selected cluster, therefore search space for Web robots was narrowed. But the search space was not well narrowed because a user doesn't interest all Web pages in a selected cluster. The method with such traditional search strategies can't navigate sufficiently for a user.

In this paper, in order to give some rules valid weights and narrow a search space, we extend the condition part of the rules to a sequence of classes of Web page types. It is important to support a user in his/her browsing task that a search space for Web robot is narrowd without gathering useless Web pages. We

propose the search strategy called "**Navigation Search**" based on some rules called "**Navigation Rules**". Supporting to reach to his/her interested Web pages efficiently by gathering more deep Web pages and displaying Web pages gathered by Web robots intelligible is very important too. We also describe a method to provide interface display Web pages gathered by Web robots at browsing task. Using this interface, a user will be able to modify crawling direction of Web robots easily.

2 Search strategy

2.1 Traditional strategies

The following main search strategies are available for gathering Web pages.

- *Breadth-first search*[7]
  Web robots use this search strategy when they gather Web pages for database used by some search engines. By using this search strategy, subjects of search Web pages is spread, and gathered Web pages will be uniform in subjects. In addition, probability that Web robots access to particular Web sites concentrically will be decreased.

- *Depth-first search*[7]
  To search some relevance Web pages, this search strategy look like browsing strategy by internet users. The behavior based on the assumption that some pages liked from a page similar to the source of page. This assumption was declare in ARACHNID[5].

- *Strategy using artificial life technique*
  Agents have energy, which is gained from relevant Web pages and lost from irrelevant pages. Agents having high energy can reproduce themselves and others having low-energy may die. Relevant Web pages is identified with this algorithm.

To navigate for a user with a current seeing Web page, agents have to gather relevance Web pages in a short term. Furthermore, as a user can predict the kind of Web pages linked a page partly, agents have to deepen the search space in the term. If above search strategies apply to agents for user's navigation, some problems will appear. Using breadth-first search, important Web pages aren't sufficiently gathered. Although, a search space for agents will be larger, and we can't reach to the depth Web pages. Using depth-first search, as user's interest changes frequently, very simillar Web pages are gathered in a few subjects. Using some techniques based on artificial life, it is very difficult to reflect user's interest to Web robots. PWM[8] integrates breadth-first search and depth-first search in order to gather Web pages interest for user's efficiently. Web robots search Web pages based on breadth-first search in PWM to gather Web pages in various subject. In order to make narrow the subject of gathered Web pages for a user, the Web pages will be divided into some clusters with SOM[3] and clusters are displayed in a window as a 2D map. A user can select clusters about which he/her wants know more, and Web robots gather Web pages that will be placed the cluster. The way to select a cluster seems like depth-first search. But a user isn't interested in all Web pages in the cluster which was selected, because a search space was not well narrowed.

For navigate to a user, we have to need novel search strategies instead of the one integrated above search strategies. We propose a novel search strategy called "**Navigation Search**" based on using rules in this paper.

2.2 Navigation Search

Many navigation systems compare keywords in a current Web page or weighted keywords in his/her profiles with keywords in the next linked Web page. But we may not gather important Web pages because of differences between Web pages in meaning. In such a case, it is important to understand correctly at the meaning of this document. On the other hand, links are selected based on the kind of Web pages in our method. Our rules are represented as the following

$$\textbf{if } p_t \in C_j^p \textbf{ then} \text{ select } l_t \left(L \cap \in C_k^l\right)$$

The t is a sequential number that describe the frequency of Web pages p selection by a user, and it is increased incrementally. The $p_{t=0}$ is current Web page a user seeing. C^p means a set of Web pages in the Web page class, C^l means a set of links in the link class, and l means a link. The L is a set of links in current Web page. The j and k is labels of the Web page classes. If some rules apply to some Web pages in browsing history and the current Web page, the next link in a current Web page is determined. Example of Web page classes C^p are shown as the following.

- *"Link Page"* : Web pages in this class include many links to others. The threshold of number of links is set beforehand.
- *"Image Page"* : Web pages in this class include many images. The threshold of number of links is set beforehand.
- *"English and Japanese Page"* : Web pages in this class written in English and Japanese half and half.
- *"Page of K"* : Web pages in this class include all keywords in one's set K.

Example of link classes C^l are shown as the following.

- *"links near an image"* : Links in this class be placed near an images.
- *"next link"* : Links in this class are the first link of the other links which aren't selected by a user yet.

These rules are used by Web robots for gathering the next Web page. By applying rules to browsing history and predict Web pages again and again, Web robots can crawl automatically. The systems using this rules will not be adapted for users' interest shift dynamically. Because the systems can't understand the kind of Web pages the user want at that time. Using these rules, many links matched these rules are selected, and search space of Web robots will be spread.

Then in order to give all rules the valid weights and narrow a search space, we extend the condition part of the rules to a sequence of Web page classes as the following.

$$\textbf{if } p_{t-i} \in C_x^p \wedge \cdots \wedge p_t \in C_z^p \textbf{ then } \text{select } l_t \in \left(L \cap C_k^l\right)$$

We called this rules "**Navigation Rules**". At the time of applying a rule, compare the condition part of the **Navigation Rule** with the sequence of user's browsed Web pages. Furthermore by compare the sequence include Web pages correspond to selected links with **Navigation Rules** again and again, Web robots gather Web pages based on the kind of Web pages a user want. If many **Navigation Rules** was matched the sequence, most weighted rule is applied at first. Weights of **Navigation Rules** are modified with feedback from a user as shown later, and we expect the weights make values relative to probability of behavior the user will do. To also approach a human behavior, the action part of rules include not only link types but also actions about 'back' and 'forward' for functions of Web browser. We called this search strategy "**Navigation Search**".

3 System's Overview

The system's overview with **Navigation Rules** for gathering Web pages shown in Fig1. The system always observe user's browsing task. **Navigation Rules** are compare with sequence of browsed Web pages each user's selecting of links, and selected links are written in the working memory. Web robots always gather Web pages from the WWW based on condition of working memory asynchronously with user's browsing task. List of links for gathering in working memory is also modifyed by user feedback. Gathered Web pages by Web robots based on matched rules provide to a user as an interface for understanding the kind of Web pages gathered easily. This interface makes a user to access interested Web pages directly.

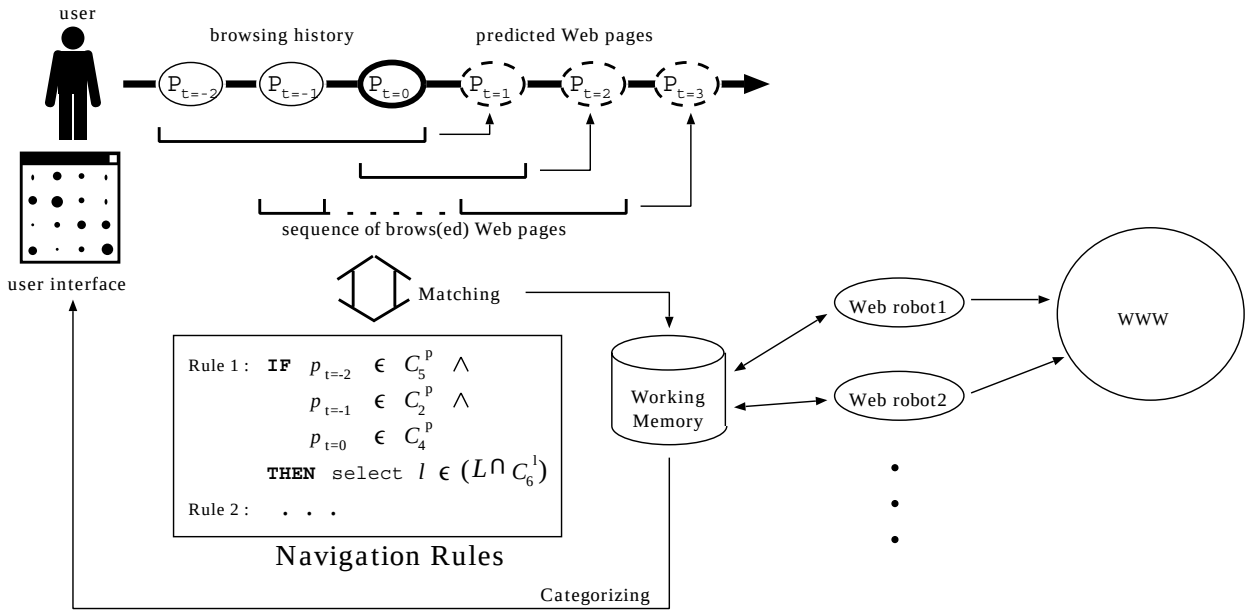

Figure 1: System's Overview

4 Adjusting weights of a rule by user feedback

In order to give all **Navigation Rules** the valid weights, weights of rules applied frequently make be high. User feedback for learning based on rules doesn't require explicit evaluation of Web pages, and it performed naturally. It is important to be decreased a load of evaluation of gathered Web pages. If links which predicted with our method are selected by a user in his/her browsing, weights of the **Navigation Rules** which determined selection of those links are increased. The weights of the other **Navigation Rules** are decreased. Some rules which value decrease under the threshold are eliminated for the cost of calculate and validity of the **Navigation Rules**. The weight values of many **Navigation Rules** which have a single Web page class in the precondition tend to be, high because such **Navigation Rules** match to browsing history frequently. Therefore **Navigation Rules** weight with value relative to the length of Web page classes in the precondition. We can expect to adapt our system for a user in changing user's interest frequently. Furthermore gathering useless Web pages will be decreased because **Navigation Rules** have valid weights, and Web robots can reach more deep level Web pages.

5 User Interface

User interface shown in the Fig1 display Web pages gathered by Web robots visually. The window of a user interface is displayed in the side of the Web browser window during browsing, and the contents indicated the interface window is updated whenever a user select a new link or a user is only looking the Web page. When some links are recommended to a user graphically by modifying original Web page contents like any other systems, if the recommendation is not valid, it is obstacle to a user. Our interface is not prevent user's browsing task, and it can use whenever the user want use. Predicted Web pages in this interface display that a user want what kind of Web pages and will reach what kind of Web pages.

There are two types of Web pages when it was displayed we think; the Web page types in condition parts of **Navigation Rules** and subjects of Web pages. For example, some Web page clusters are displayed for each subjects at the first, and Web pages displayed for each class of condition part in **Navigation Rules** ordering by the weights. Web pages are divided into some clusters based on SOM[3]. Clusters are displayed in an interface window with some characteristic keywords for the clusters. If a user select a cluster, Web pages in the cluster ware displayed for each Web page classes with the rate that the Web pages locate into

the cluster. A user can understand predicted Web pages easily using by this interface. If a user find an interested Web page, he/she can see the Web page directly by selecting the link of the Web page.

6 Conclusions

In this paper, we defined the rules called "**Navigation Rules**" using sequence of browsed Web pages instead of only access log of browsing, and we proposed the way to predict links and actions which the user will do. And we described the way to gather Web pages for navigation by Web robots based on **Navigation Rules** called "**Navigation Search**", and then described the way to construct a user interface that can use whenever a user want. The characteristic of our interface is that a user can give feedback without prevent his/her browsing tasks. Our system can navigate a user effectively by narrowing a search space for Web robots and gathering Web pages based on the user's interest.

References

[1] P. De Bra and R. Post. Information Retrieval in the World-Wide Web: Making Client-based Searching Feasible. *Computer Networks and ISDN Systems*, 27(2):183–192, 1994.

[2] T. Joachims, D. Freitag, and T. Mitchell. WebWatcher: A Tour Guide for the World Wide Web. In *Proceedings of the Fifteenth International Joint Conference on Artificial Intelligence*, pages 770–775, 1997.

[3] T. Kohonen. *Self-Organizing Maps*. Springer, Berlin, Heidelberg, 1995. Second Extended Edition 1997.

[4] H. Lieberman. Letizia: An Agent That Assists Web Browsing. In *Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence*, pages 924–929, 1995.

[5] F. Menczer. ARACHNID: Adaptive Retrieval Agents Choosing Heuristic Neighborhoods for Information Discovery. In *Machine Learning: Proceedings of the Fourteenth International Conference*, pages 227–235, July 1997.

[6] F. Menczer, R. K. Belew, and W. Willuhn. Artificial Life Applied to Adaptive Information Agents. In *Working Notes of the AAAI Symposium on Information Gathering from Distributed, Heterogeneous Databases*. AI Press, 1995.

[7] S. Russell and P. Norvig. *Artificial Intelligence –A Modern Approach–*. Prentice-Hall, 1995.

[8] S. Yamada and N. Nagino. Constructing a Personal Web Map with Anytime-Control of Web Robots. In *Conference on Cooperative Information Systems*, pages 140–145, 1999.