A Novel Method for Crawler in Domain-specific Search

Chunxia YIN1, Jian LIU2†, Chao YANG1, Huiying ZHANG1

1 College of Ocean, Hebei Agricultural University, Qinhuangdao 066003, China
2 Hebei Vocational College of Foreign Languages, Qinhuangdao 066311, China

Abstract

A focused crawler is a Web crawler aiming to search and retrieve Web pages from the World Wide Web, which are related to a domain-specific topic. Rather than downloading all accessible Web pages, a focused crawler analyzes the frontier of the crawled region to visit only the portion of the Web that contains relevant Web pages, and at the same time, try to skip irrelevant regions. In this paper, we present a new crawling strategy that employed the C4.5 decision tree to predict the relevance of a link target, and combined the algorithm with the link characteristic of parent pages. Experimental results indicate that the new crawling method has better performance, and it was able to fetch higher topic relevant information.

Keywords: Focused Crawler; Domain-specific; C4.5 Decision Tree

1. Introduction

Nowadays, the World Wide Web has become the largest hypertext system in existence, providing an extremely rich collection of information resources, and more and more people use the Web for information searching. Researchers have proposed many techniques to facilitate the information seeking process on the Web. Search engines play a very important role in information searching and management on the Web. However, with the explosive growth of the Web, it is becoming increasingly difficult for search engines to keep their services effective. On the one hand, as the number and size of stored documents is growing even faster and site contents are getting updated more and more often. General search engine cannot grow that fast and therefore they cover an ever-decreasing segment of the Web. The large size and general focus of their indices entails a rather low precision. On the other hand popular topical pages cannot fit every user’s interests equally well. Therefore, the generic search engines will become less appealing to a user with specific information needs that do not fall into the most popular categories. To address the above problems, many domain-specific search engines that keep their Web page collection only in particular domains have been built. Contrary to general search engines, domain-specific search engines are inexpensive in storage requirements and more appropriate to services catering for specialty markets and target groups.

Domain-specific search engines use focused crawlers [1] to selectively collect Web pages relevant to particular domains such that they can keep smaller Web page collections and, at the same time, provide

† Corresponding author.

Email addresses: chinafather@163.com (Jian LIU), lovelyfatbear@163.com (Chunxia YIN).

1553-9105/ Copyright © 2009 Binary Information Press
December, 2009
search results with high precision. Unlike general-purpose Web crawler which automatically traverses the Web and collects all Web, focused crawling is designed to gather collection of pages on specific topic. A focused crawler tries to “predict” whether or not a target URL is pointing to a relevant and high-quality Web page before actually fetching the page, and then it follows the most appropriate links, leading to retrieval of more relevant pages and greater saves in resources.

There are many previous researches focusing on algorithms of Web page crawling. The main purpose of those algorithms is to gather as many relevant Web pages as possible, and most of them emphasis on how to collect more topic relevant information. In this paper, we presented a new method which employs some present algorithms. Experimental results indicate that the new crawling method has better performance. which was able to fetch higher quality relevant information. This paper is organized as follows: Section2 presents the related work in focused crawling. Section 3 proposes the crawling strategy Section 4 gives evaluations about our crawling strategy Section 5concludes the paper.

Related work. In recent time there has been much research in focused crawler. The emphasis is the algorithm for collecting Web pages related to a particular topic. The fish-search algorithm for collecting topic-specific pages is initially proposed by P. DeBra et al. [2]. In Fish-Search, the Web is crawled by a team of crawlers, which are viewed as a school of fish. If the "fish" finds a relevant page based on keywords specified in the query, it continues looking by following more links from that page. If the page is not relevant, its child links receive a low preferential value. The crawler dynamically maintains a list of uncollected URLs that ordered by priority of being gathered and downloads the Web pages according to the queue. In the processing, these URLs in the relevant Web pages are given higher priorities than the others in the irrelevant Web pages.

Based on the improvement of fish-search algorithm, M. Hersovici et al. proposed the shark-search algorithm [3]. The effects of those characters in the hyperlinks are considered when the priority of URLs being calculated in this algorithm. At the same time, the vector space model is introduced to calculate the similarity factor of Web pages. At the same time, the vector space model is introduced to calculate the similarity factor of Web pages.

A popular approach for focused resource discovery is the best-first search algorithm where two URL queues are maintained; one containing the already visited links (from here on AF) and another having the, yet unvisited, references of the first queue, also called crawl frontier (from here on CF)[4]. The challenging task is periodically reordering the links in the CF efficiently. The importance metrics can be either interest driven where the classifier for document similarity checks the text content or popularity/location driven where the importance of a page depends on the hyperlink structure of the crawled document.

More recently, link and social network analysis have been applied to Web IR to identify authoritative information sources [5]. Here, the impact factor corresponds to the ranking of a page simply by a tally of the number of links that point to it, also known as backlink (BL) count or in-degree. But BL can only serve as a rough, heuristic-based, quality measure of a document, because it can favour universally popular locations regardless of the specific query topic. PageRank (PR) is a more intelligent connectivity-based page quality metric with an algorithm that recursively defines the importance of a page to be the weighted sum of its backlinks’ importance values [6]. An alternative but equally influential algorithm of modern hypertext IR is HITS, which categorizes Web pages to two different classes; pages rich and relevant in text content to the user’s query (authorities) and pages that might not have relevant textual information but can
lead to relevant documents (hubs). Hubs may not be indexed in a vertical engine as they are of little interest to the end user, however both kind of pages can collaborate in determining the visit path of a focused crawler.

The neural network is adopted in the Info Spider designed by F. Menczer et al.[7]. The information in the hyperlinks of Web pages is extracted as the input of the neural network, and the output is considered as the guidance to farther collection. The VTMS (Web Topic Management System) crawler, designed by S. Mukherjes[8], downloads the seed URLs that relevant to the topic and creates a representative document vector (RDV) based on the frequently occurring keywords in these URLs.

The Context Graph method, proposed by Diligenti et al. [9] uses backlinks to estimate the link distance from a page to target pages. Their method starts from a set of seed documents, follows backlinks to a certain layer, and then builds up a context graph for the seed pages. A classifier is constructed for each layer using the Naive Bayes algorithm. As a new document is found, it is classified into a layer. Documents classified into layers closer to the target are crawled first. The experiments showed that this approach maintained a higher level of relevance in the retrieved Web pages. However, the assumption that all pages in a certain layer from a target document belong to the same topic described by a set of terms does not always hold.

All these works present algorithms that enable their crawlers to select Web pages related to a particular topic. The main purpose of those algorithms is to gather as many relevant Web pages as possible. Therefore, the efficiency of topic-specific Web crawling is measured in terms of a proportion of the number of relevant Web pages and the total number of downloaded Web pages. If this proportion is high, it means that the crawler can collect more relevant Web pages than irrelevant ones. This is a result we expect for the Web crawler.

2. Crawling Strategy

2.1. Topic Specification

There are two basic approaches to specify user interest in topical crawler: taxonomy-based and keyword-based. In taxonomy-based approach, users select their interest from topics of a predefined taxonomy. This approach is simple for users to select their topics. However, it put a great limitation on the set of possible topics. It might work well for topic-specific search engines, but is insufficient for personal crawlers whose objective is to collect all documents of personal interest. In keyword-based approach, interest is specified by keywords that define the targets of the interest. It is more flexible than taxonomy-based approach. However, users might not know how to specify their queries precisely, and sometimes even might not be clear about what their true targets are, especially when they are not familiar with the domain.

Specification of a topic consists of a set of target examples and a set of keywords. Our Crawler use keywords to query search engines such as Google to find target examples, and then it extracts and ranks terms from target exampled pages using $\chi^2$ statistics measure that is shown to be stable and effective to select informative terms in text categorization [10]. The top K informative terms are sorted and suggested to users as the set of keywords to specify the topic.
2.2. Relevance Measure

When a Web page has been collected, the crawler should have relevance judgment to decide which Web page is relevant, i.e. it is related to a topic of user interest. There are many relevance judgment techniques. For instance, Chakrabarti et al. [1] use classification technique to build a set of example Web pages, which are both relevant and irrelevant to the interested topic, to constitute the topic model. Classifier will compare a candidate Web page with the topic model to justify whether it is related to the topic. There are many ways to compute the topic similarity, but here we only focus on a vector space model. The main idea of the vector space model is to represent a Web page as a vector.

Each distinct word in that page is considered to be an axis of a vector in a multi-dimensional space. The direction of a vector corresponds to the content of the Web page. Two Web pages are assumed to be relevant, i.e. relating to the same topic, mathematically, if two vectors point to the same direction, this means that they have an angle of zero degree and their cosine value becomes 1. Therefore, a vector matching operation, based on the cosine correlation, is used to measure the degree of topic similarity between the page and the interested topic. We define \(sim(k, d)\) to be the page similarity function with the formula written in (1).

\[
sim(k, d) = \frac{\sum_{t \in (k \cap d)} f_{kt} f_{td}}{\sqrt{\sum_{t \in k} f_{kt}^2} \sum_{t \in d} f_{td}^2}
\]

Here, \(k\) is a set of keywords describing an interested topic, \(d\) is the Web page which we want to compare, \(f_{kt}\) and \(f_{td}\) are the number of terms \(t\) in the set of keywords. \(k\) and a Web page \(d\), respectively. The range of \(sim(k, d)\) value lies between 0 and 1, and the similarity increases as this value increases.

2.3. URL Ordering Method

T. T. Tang built a focused crawler for the mental health topic of depression, which was able to selectively fetch higher quality relevant information [11]. In their crawler, they used the C4.5 decision tree [12] to predict the relevance of a link target. In our crawler, we also employed the C4.5 decision tree to predict the relevance of a link target. The classifier is based on words in the anchor text, words in the target URL and words in the 100 characters before and after the link (link context). If we found multiple links to the same URL, we included all available anchor contexts. This is a relatively standard approach. To produce a confidence score at each leaf node of the decision tree we used a Laplace correction formula [13]:

\[
\text{confidence}_\text{level}_i = \frac{N_k + \lambda}{N + \sum_{k=1}^{K} \lambda_k}
\]

where \(N\) is the total number of training examples that reach the leaf; \(N_k\) is the number of training examples from class \(k\) reaching the leaf; \(K\) is the number of classes and \(\lambda_k\) is the prior for class \(k\) and is usually set to be 1. In our case, \(K\) is 2 because we only had two classes, relevance and irrelevance.

The overall score for a URL was given by:

\[
\text{URLScore} = \text{confidence}_\text{level}_i \times \sum_{i=1}^{m} \phi(i) \times D\text{Score}(i) / m
\]
\[ \phi(i) = \frac{t^r + 1}{t + \theta} \]  \hspace{1cm} (4) \\
\[ \text{DScore}(i) = \text{sim}(i, \text{topic}) \]  \hspace{1cm} (5)

\( t^r \) denotes the number of the topic relevant pages in all the son pages of page \( i \). \( t \) is the total number of the crawled son pages of \( i \). \( \theta \) is the factor parameter and it is often set as 2. \( \text{DScore}(i) \) is computed using (1), and it denotes the topic relevant score of the page \( i \).

3. Evaluation

3.1. Experimental Setup

In this section we will present our experiences with focused crawler. Three crawling strategies were implemented for performance comparison: Breadth first, Best first, and our crawling strategy which we called “BBF”.

The evaluation methodology commonly used in information retrieval is to calculate the recall and precision. But in topic-driven Web resource discovery, it’s impossible to get the number of topic relevant pages online, so recall becomes unavailable. Precision represents how much percent of all fetched pages is topic relevant, and we employed the precision metric to evaluate the crawled pages. Precision often called 'harvest rate' is used as a major performance metric in the focused crawler community. Harvest rate is defined as the percentage of the crawled pages that are relevant to the specified topic:

\[ \text{harvest rate} = \frac{\text{relevance pages}}{\text{pages downloaded}} \]  \hspace{1cm} (6)

In order to evaluate our crawler algorithms, we need topics and some corresponding relevant seed URLs. Four topics, “sport”, “Chinese sport”, “music”, and “American music” respectively, are chosen for the experiments. Target topics were defined in keywords; Keywords were used in Google search. The result pages were parsed to extract the hit URLs, which were presented to the system as seed URLs. For each topic, crawlers started with 10 seed sample URLs, and crawled about 1100 Web pages.

3.2. Performance Description

Table 1 shows the final harvest rates of four topics after crawling 1100 pages, and it displays that our crawling strategy “BBF” outperformed Best first and Breadth first in all topics.

The harvest rate for the topic “sports” with relevance is shown in Fig. 1, the topic “Chinese sports” in Fig.2, the topic “music” in Fig. 3 and “American music” in Fig. 4. In our experience, Breadth-First without judging on the context of the unvisited URLs, performed not well. It depends heavily on the localization of the relevant pages and Web sites. Best-First predicts the relevance of the potential URLs by referring to the whole context of the visited Web page. All out-links in one page have same priority. It only grouped the unvisited URLs based on the page picked up from, and there is no difference within each group. So it has low accuracy when there is a lot of noise in the page or the page has multiple topics, as showed in topic “American music”. In all topics, the proposed BBF algorithm performs the best, and presents stable performance.

4. Conclusion

In this paper, we present a new crawling strategy which we called “BBF”, it employed the C4.5 decision tree to predict the relevance of a link target, and then combined it with the link characteristic of parent
Our experiments have shown that the focused crawler using the BBF strategy has better performance than the Breath first and Best first algorithms. However, there are some limitations in our study, e.g. the volume of examined Web pages is too small. We expect to do more extensive test with large volume of Web pages, later. In future, some specifically automatic classifier would be used to judge the relevance of crawled Web pages.

Table 1 The Average Harvest Rates of Different Topics

<table>
<thead>
<tr>
<th>Topic/Algorithm</th>
<th>Breadth first</th>
<th>Best first</th>
<th>BBF</th>
</tr>
</thead>
<tbody>
<tr>
<td>sport</td>
<td>0.183</td>
<td>0.519</td>
<td>0.720</td>
</tr>
<tr>
<td>Chinese sport</td>
<td>0.180</td>
<td>0.478</td>
<td>0.703</td>
</tr>
<tr>
<td>music</td>
<td>0.222</td>
<td>0.553</td>
<td>0.751</td>
</tr>
<tr>
<td>American music</td>
<td>0.173</td>
<td>0.450</td>
<td>0.662</td>
</tr>
</tbody>
</table>

Fig.1 The Harvest Rate on Topic “Sports”.  
Fig.2 The Harvest Rate on Topic “Chinese Sports”.  
Fig.3 The Harvest Rate on Topic “Music”.  
Fig.4 The Harvest Rate on Topic “American Music”.

References


