Systems and Computers in Japan, Vol. 38, No. 2, 2007
Translated from Denshi Joho Tsushin Gakkai Ronbunshi, Vol. J89-D, No. 2, February 2006, pp. 199–209

A Method for Language-Specific Web Crawling and Its Evaluation

Takayuki Tamura,[1,2] Kulwadee Somboonviwat,[2] and Masaru Kitsuregawa[2]

[1]Information Technology R&D Center, Mitsubishi Electric Corporation, Kamakura, 247-8501 Japan

[2]Institute of Industrial Science, The University of Tokyo, Tokyo, 153-8505 Japan

SUMMARY

Many countries have created Web archiving projects aiming at long-term preservation of Web information, which is now considered precious in cultural and social aspects. However, because of its borderless character, the Web poses obstacles to comprehensively gathering information originating in a specific nation or culture.

This paper proposes an efficient method for selectively collecting Web pages written in a specific language. First, a linguistic graph analysis of real Web data obtained from a large crawl is conducted in order to derive a crawling guideline, which makes use of language attributes per Web server. The guideline then is formed into a few variations of link selection strategies. Simulation-based evaluation reveals that one of the strategies, which carefully accepts newly discovered Web servers, shows superior results in terms of harvest rate/coverage and runtime efficiency. © 2007 Wiley Periodicals, Inc. Syst Comp Jpn, 38(2): 10–20, 2007; Published online in Wiley InterScience (www.interscience.wiley.com). DOI 10.1002/scj.20693

Key words: Web archiving; focused crawling; language identification; Web graph; crawling simulation.

1. Introduction

While the enormous amount of digital documents on the Web are recognized as precious information sources in terms of cultural and social aspects, a sense of crisis concerning their disappearance is intensifying. To deal with this situation, a number of countries have formed Web archiving projects, mainly under the leadership of national libraries, aiming at long-term preservation of their cultural heritage [1]. The National Diet Library Japan started an experimental project WARP (Web ARchiving Project) in 2002 [2]. Initially, these projects restricted their target Web sites, due to the cost of retaining the quality of collections and to legal issues. Although these projects tend to switch to comprehensive crawling, their crawling ranges are still limited to country-specific domains (ccTLD, country code top level domains: e.g., .jp for Japan, .th for Thailand), with the exception of the Scandinavian countries and the Internet Archive [3], a nonprofit organization in the United States.

As the Web is borderless and ccTLD is merely an option, commercial activities tend to be lively in transnational domains such as .com. Consequently, for the purpose of analyzing social phenomena from the real lives of individuals, it is essential to capture people's activities extending outside the country's domain. This paper proposes a method for selectively collecting Web pages written in a specific language without the help of domain names. The effectiveness of the method is evaluated through simulations using real Web data.

Contract grant sponsor: Comprehensive Development of e-Society Foundation Software program of the Ministry of Education, Culture, Sports, Science and Technology of Japan.

© 2007 Wiley Periodicals, Inc.

Language-specific Web crawling is particularly important for countries with national languages that are not widely used on the Internet. The Internet Archive mentioned above and search engine companies are conducting comprehensive crawls over the entire Web. Since this will result in a collection primarily in English, it suffices for organizations in the English-speaking world to postfilter a tiny fraction of the collection written in unwanted languages. Contrarily, for the world of minority languages, the postfiltering approach is unacceptable because it involves a great waste of computational and network resources for eventually unwanted data. What is needed is an efficient Web crawling method which is able to match the resource requirements to the volume of target Web pages.

Chakrabarti and colleagues proposed focused crawling, and originated a series of investigations of methods for selectively collecting topic-specific Web pages [4–6]. A focused crawler employs a pretrained classifier to categorize the content of a downloaded Web page. Links found in the Web pages which belong to the relevant categories are followed with high priority, while links in other pages are discarded in the hard focused mode, one of two modes of the focused crawler. The other mode, the soft focused mode, lets the crawler follow the latter links with low priority. The objective of the soft focused mode, which makes use of irrelevant pages, is to avoid the stagnation which occurs when relevant pages are not connected directly with each other.

Language-specific Web crawling runs in the same framework as focused crawling, with the topical classifier replaced with a language identifier which indicates whether a downloaded Web page is written in the target language. According to the judgment, links extracted from the Web page will be followed with a corresponding priority, or discarded. In general, it is not easy to accurately identify the language or the character encoding scheme of a given text. As a solution to this, a statistical method based on n-grams of byte sequences has been proposed [7, 8]. The method first builds a language model for known languages and character encoding schemes from the occurrence frequency of each n-gram in the language's text corpus, and then compares the input text to each of the language models to determine the most similar one. In addition, a language identification capability is often seen in Web browsers, which need to properly render Web pages in various languages. Some of them provide components reusable in external applications, such as the `universalchardet` component [10] of Mozilla [9], and the `MLang` component of Microsoft Internet Explorer.

We are also aiming at substantial improvement of efficiency by identifying the language at the server level, with coarser granularity than the page level. Unlike the topics with which focused crawling methods deal, we can naturally assume that the language used in a server is uniform. Analysis of the graph structure of actual Web data confirmed the locality assumption with regard to server languages, which formed the basis of the proposed crawling method.

Baeza-Yates and colleagues [11] describe a method for country-level Web crawling, but the focus of this work is on scheduling issues under network resource constraints, and their crawling targets are based on country-level domains (i.e., .gr and .cl). In addition, Ghani and colleagues [12] propose an approach to building corpora of minority languages from Web information. Their method queries Internet search engines with terms peculiar to the target language, and feeds the results back to the next queries to purify the corpus. However, the method aims at obtaining a small set of documents, and is not applicable to archiving, where comprehensive information in a specific language is to be collected while discovering information resources on the Web.

The rest of this paper is organized as follows. Section 2 describes the Thai Web data set as a target for evaluation, and the language identification result of its Web pages. In Section 3, we point out the characteristics of the Thai Web data set in terms of graph structure and language distribution. Section 4 explains the proposed crawling method, in particular its link selection strategies based on per-server language identification. In Section 5, based on the simulation-based evaluation results, we discuss the merits of the proposed strategies in terms of the quality of collected pages and runtime efficiency. Section 6 concludes the paper.

2. The Data Set and Its Linguistic Characteristics

2.1. Target language

For ease of evaluation of various crawling strategies, we first made a local copy of a subset of the Web space. A data set appropriate for testing language-specific Web crawling strategies should contain not only Web pages in the target language but also sufficient Web pages in other languages. In this research, we chose Thai as the target, because it is not dominant on the Web.

We gathered a subset of Web pages in and around Thailand by following HTML links[†] recursively from the seed URLs below, which are the home pages of some popular Web sites in Thailand.

- http://www.matichon.co.th/
- http://www.sanook.com/
- http://www.siamguru.com/

[†] We ignored JavaScript in the Web pages.

Our crawler ran for about 1 month (from July to August 2004) and produced a data set of 18,344,127 HTML documents downloaded from 574,111 servers. There were also 360,404 non-HTML documents, which we ignored in the evaluation in Section 5. The proportions of the number of Web pages by Internet domain were: .com 56.3%, .de 7.9%, .jp 7.6%, .org 7.5%, .net 6.9%, and .th 4.7%.

Since we were crawling Thai Web from Japan on a large scale, we had to observe appropriate etiquette by accessing each server at intervals of 1 to 2 minutes, whereas up to 30 different servers were accessed simultaneously. As a result, the resulting data set is no longer consistent with the range covered by the theoretical breadth-first search. In addition, about 20 million URLs remained uncrawled in the queue (frontier) when we stopped the crawling. Thus, it should be noted that part of the data set with large distances from the seeds does not reflect the real Thai Web accurately.

2.2. Language identification of Web pages

The procedure we used for identifying the language of a Web page (an HTML document) is as follows.

1. Parse the HTML to extract character encoding information from the META element if any. For example, the following META element specifies "TIS-620" as a charset name.

```
<META http-equiv="Content-Type" con-
  tent="text/html; charset=TIS-620">
```

Note that an extracted character encoding name is adopted only if it is valid[†] and not "ISO-8859-1," which we regard as noise because it tends to be set unconditionally by some HTML authoring tools.

If the META elements for character encoding specification appear repeatedly in the same page, the last occurrence will take precedence.

2. If the valid character encoding name is missing, after removing all the HTML tags from the Web page contents, we submit the remaining text to TextCat,[‡] a language guesser based on n-gram statistics.

In cases where TextCat is not able to exactly determine the language of a given text, it will return a concatenation of candidate language names, or occasionally the string "unknown."

3. When the above steps yield "UTF-8," a language-neutral character encoding, an attempt is made to convert the text of the page into TIS-620 and, unless the conversion fails, it is submitted (again) to TextCat to confirm whether the text is in Thai (i.e., plain ASCII).

Table 1. Language identification result of the data set (in number of pages)

Languages	Domains		
	.th	Others	Total
Thai	591,683 3.2%	1,131,088 6.2%	1,722,771 9.4%
English(ASCII)	108,149 0.6%	7,971,458 43.4%	8,079,607 44.0%
Others	155,628 0.9%	8,386,121 45.7%	8,541,749 46.6%
Total	855,460 4.7%	17,488,667 95.3%	18,344,127 100%

Though the charset name can also appear in the Content-Type field of an HTTP header, we ignore such specifications, which are not trustworthy in general.

According to the above procedure, a Web page is identified as a Thai page (i.e., relevant to the crawl) in either of the following cases:

- The META element's character encoding name is either "TIS-620" or "Windows-874."
- The output returned by TextCat includes the string "[thai]."

2.3. Result of language identification

Table 1 shows the result of language identification of the Web pages in the data set. As we can see, less than 10% of the data set are written in the Thai language, so the data set is suitable for evaluating Thai language-specific Web crawling strategies. In addition, since only one-third of the Thai pages belong to the .th domain, a domain-specific crawling method would miss too many Thai pages. As for server statistics, we found that only 26,671 servers (4.6%) in the data set contained at least one Thai page. These observations indicate that it is important for Thai language-specific Web crawling to discover servers which contain Thai pages (referred to as Thai servers hereafter) without relying on domain names.

Tables 2 and 3 show the verification results of META-based and n-gram-based language identification methods,

Table 2. Verification result of META-based language identification method (in number of pages)

Guessed languages	Actual languages		
	Non-Thai	Thai	Total
Non-Thai	57	5	62
Thai	59	285	344
Total	116	290	406

[†] i.e., if they can be found in major character set names and aliases defined by IANA (Internet Assigned Numbers Authority) [13].

[‡] TextCat was originated by Cavnar and Trenkle [7], and some independent implementations are available. We used the C-library implementation, libTextCat [14].

Table 3. Verification result of n-gram-based language identification method (in number of pages)

	Actual languages		
Guessed languages	Non-Thai	Thai	Total
Non-Thai	363	14	377
Thai	0	458	458
Total	363	472	835

respectively, using small samples identified by either of the methods. According to Table 2, as much as 17.2% (= 59/344) of Web pages with Thai character encodings in META elements are actually written in non-Thai languages such as English. However, about half of these pages (27 pages) are English pages on Thai company sites, which contain Thai content elsewhere. In practice, these sites will actually have a positive effect on coverage of Thai information collection. If we regard Thai-company-originated English pages as relevant, the precision of META-based Thai language identification becomes 90.7%. About 20 out of the 32 remaining non-Thai pages seemed to be mirror pages of hotel information sites aiming at SEO (Search Engine Optimization) effects. META-based identification would be improved further by employing detection techniques for such mirror pages/servers.

Due to rare pages which were written in Thai but specified non-Thai character encodings, the recall of META-based identification was 98.3% (= 285/290). These pages used numeric notation of Thai characters in Unicode (e.g., `&#3627;`). Upon detecting such notation, the page's text could be converted into a Thai character encoding and submitted to TextCat as in the case of UTF-8 to avoid misses.

The n-gram-based Thai language identification method achieved superb results in both precision (100%) and recall (97.0%).

Note that 86.7% of all Thai pages (about 1.7 million as shown in Table 1) were identified by the META-based method. Though the aforementioned language identification procedure attempts the META-based method first for runtime efficiency, consulting the result of the n-gram-based method as well would improve the precision of language identification further, at least for Thai. In this paper, we leave the improvement of language identification performance as future work and focus on the crawling strategies that obey the output of the language identifier.

3. Graph Structure of the Data Set

3.1. Thai Web graph

In this section, we describe the characteristics of the Thai Web graph derived from the pages and links in the data set. The graph consists of 39,078,795 vertices, 19,953,318 of which correspond to uncrawled pages (only their URLs are known from the link destinations). Each vertex is labeled as Thai, non-Thai, or uncrawled according to the status of the corresponding Web page. The number of directed edges in the graph is 123,836,342, excluding duplicates and loops.

3.2. Thai page ratio with distance from the seeds

Figure 1 shows the cumulative number of pages and the cumulative ratio of Thai pages in the subgraphs of the Thai Web graph, where each subgraph is represented by the maximum distance from the three seed vertices. Note that the distance, the minimum path length in the graph, does not completely agree with the length of the path followed in the actual crawl, because subtleties such as etiquette control made the crawl different from the theoretical breadth-first search.

Though the cumulative number of pages (the size of the subgraph) rises rapidly with distance, it remains almost constant beyond a distance of 10. This implies that the data set does not contain enough data in the region due to interruption of crawling.

The Thai page ratio starts to drop rapidly from a distance of 1. This suggests that a distance-limited breadth-first strategy cannot achieve a sufficient harvest rate of relevant pages.

3.3. Distance between nearest Thai pages

Figure 2 shows the distribution of distances between a Thai page and another Thai page linked to the former via the shortest path in the Thai Web graph. A distance of 1 is for a Thai page linked directly to another Thai page. Dis-

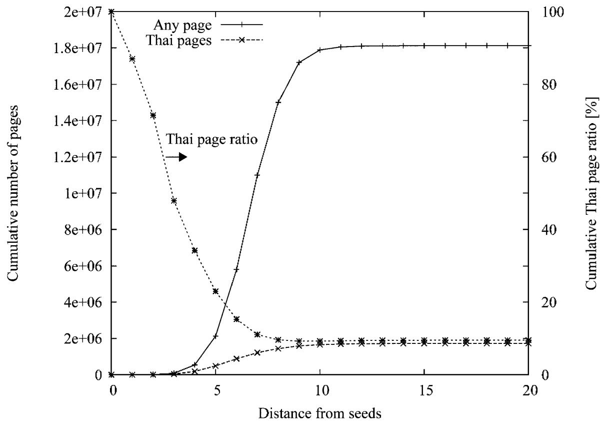


Fig. 1. Cumulative number of pages and Thai page ratio with distance from the seeds.

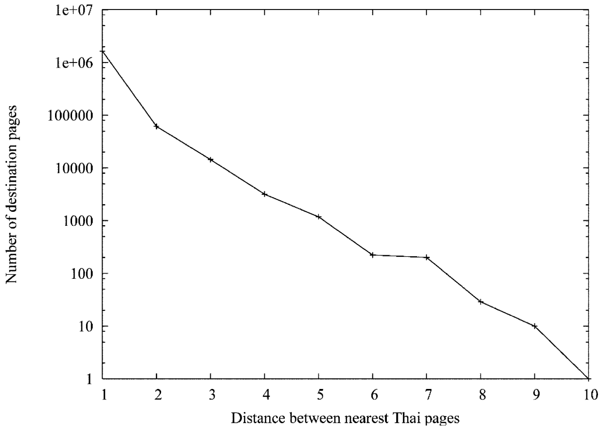


Fig. 2. Distribution of distances between nearest Thai pages.

tance > 1 means that it is necessary for a Thai page to pass through at least (distance – 1) non-Thai pages before reaching other Thai pages. As expected, most Thai pages are linked directly to other Thai pages. The number of destination Thai pages decreases exponentially as the distance increases. Nevertheless, the farthest pair of Thai pages is at a distance of 10. Hence, it should be recognized that some Thai pages cannot be obtained without downloading some non-Thai pages.

We found that the actual case of distance 10 was exceptional in that all the pages in the path were English pages in a single server (of a Thai company) with both ends labeled with the Thai charset in the META elements. On the other hand, in one of the cases with a distance of 9, a page of a Thai-related portal site in Japan (tengmoo.web.infoseek.co.jp) led to a page of a Thai hospital site (www.samitivej.co.th) via several pages in Thai information sites for Japanese. The hospital site provides content in three languages, Thai, English, and Japanese, with the home page in English. Because the link between English and Thai was implemented with JavaScript, which was ignored in the crawl, Thai pages in the site were reachable only via lengthy detours. Moreover, frequent use of HTML frames in the intermediary pages made the effective distance longer.

Thorough inspection of distant pairs of Thai pages is beyond the scope of this paper, but if the cases like the latter are common, improvement in crawler implementation, such as handling of non-HTML links and differentiation between ordinary links and frame sources, would be necessary.

3.4. Languages of source and destination pages

Tables 4 and 5 show the relationship between the languages of the source pages and destination pages for all links in the Thai Web graph. In Table 4, we excluded links to the uncrawled pages, whose languages are unknown. In Table 5, "Thai servers" denotes servers with at least one Thai page, and "non-Thai servers" denotes servers with no Thai page but with at least one non-Thai page. Servers from which we could not get any page are excluded. We can observe from Table 4 that about 70% of links from Thai pages point to other Thai pages. However, most of the links to Thai pages are intraserver links, and the percentage of interserver links from the Thai pages to other Thai pages is 37.4%. Furthermore, if non-Thai pages reside on Thai servers, 20 to 30% of the links from such pages lead to other Thai pages. Otherwise, non-Thai pages will seldom lead to Thai pages.

From the above observation, we derived the crawling strategies described in the next section, where language attributes of servers rather than pages play an important role.

Table 4. Relation between languages of link source pages and destination pages (in number of links)

	Destination (crawled pages only)	
Source	Thai page	Non-Thai page
Thai page	23,980,308 71.8%	9,428,677 28.2%
	3,833,048 37.4%	6,421,570 62.6%
Non-Thai page (in Thai server)	1,542,513 21.5%	5,625,911 78.5%
	791,862 35.3%	1,452,123 64.7%
Non-Thai page (in Non-Thai server)	74,094 0.2%	48,137,407 99.8%
	74,094 0.3%	24,731,356 99.7%

(Intra-server links are excluded in lower rows.)

Table 5. Relation between languages of link source servers and destination servers (in number of links)

	Destination	
Source	Thai server	Non-Thai server
Thai server	41,653,154 87.4%	6,013,026 12.6%
	6,886,986 53.4%	6,013,026 46.6%
Non-Thai server	286,429 0.5%	61,175,697 99.5%
	286,429 1.1%	25,655,607 98.9%

(Intra-server links are excluded in lower rows.)

4. Strategies for Language-Specific Web Crawling

4.1. Link selection based on the destination server's language

Focused crawling methods employ strategies to choose (or prioritize) links to follow among all the links found on the downloaded pages. The granularity of selection can be classified roughly into the following two types:

1. Select links in a page as a group based on per-page attributes.
2. Select links in a page one by one based on per-link attributes.

The per-page attributes include the textual features of the page, including the language. In this approach, downloading is necessary to acquire the page's attributes. Consequently, every time a set of links is discarded, downloading of an irrelevant page has already taken place. Thus, this approach is not adequate for nationwide crawling, because the ratio of relevant pages falls rapidly as the crawl proceeds, as shown in Fig. 1 of Section 3.

The latter, per-link attributes, includes the features of the anchor and surrounding text, the position where the link appears in the page, the structural features of the HTML tags around the link, and so on. Heuristic-based and learning-based approaches with these attributes can be found in the literature [5, 6].

We propose an approach based on the destination URL of each link in addition to the page's language, as a method suitable for comprehensive crawling of pages written in a specific language. In this approach, if a link's destination URL is known to belong to an irrelevant server, that link is discarded without downloading because the destination page will hardly contain links to relevant pages or servers (0.2 or 0.5%, respectively, according to Tables 4 and 5). We assume that the loss of rare links from irrelevant servers to relevant pages or servers can be compensated by collecting enough pages and links from relevant servers. Thus, this approach can reject a large number of links to (presumably) irrelevant pages on irrelevant servers without downloading them, avoiding degradation of efficiency in large-scale crawling.

Identification of relevant servers and irrelevant servers is crucial to this approach. Because Web servers keep emerging and disappearing, we need to discover relevant servers during crawling rather than enumerate them beforehand. Although relevant servers can be established as soon as their pages turn out to be relevant, that is not the case with irrelevant servers. That is, irrelevant pages do not necessarily imply that the corresponding servers are irrelevant, because relevant servers can have pages in different languages such as English, or pages with no actual content such as HTML frames. Thus, we allow up to a certain number of irrelevant pages to be downloaded from a server before we identify the server as irrelevant. Let $N_a(i)$ and $N_r(i)$ be the number of downloaded pages and the number of relevant pages, respectively, for a server i, and let T denote the tolerance for irrelevant pages per server. The identification status of server i is given as follows.

- Server i is relevant if $N_r(i) > 0$.
- Server i is undecided if $N_r(i) = 0$, $N_a(i) \leq T$.
- Server i is irrelevant if $N_r(i) = 0$, $N_a(i) > T$.

Figure 3 shows the processing of downloaded pages in the proposed crawling method.

When a newly downloaded page turns an undecided server into irrelevant, there will be other URLs of the same server in the queue. Such URLs must be removed at this point, or skipped at downloading time.

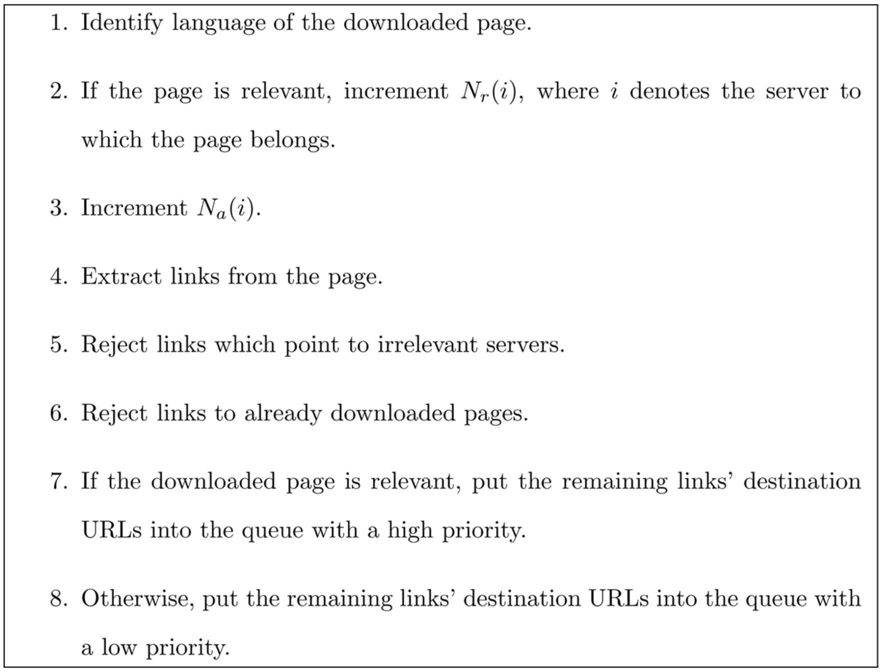
4.2. Conservative discovery of new servers

The basic strategy described above makes use of links from irrelevant pages to avoid missing relevant pages. As shown in Table 4, irrelevant pages contain many links to relevant pages and servers if they are on relevant servers. Contrarily, links from irrelevant pages on irrelevant servers should have been discarded. Once a server is established as irrelevant, it will no longer yield links. Thus, the links that need further restriction are those that originate on irrelevant pages of undecided (not yet irrelevant) servers. However, the basic strategy in the previous section does not differentiate undecided servers from relevant servers, in that irrelevant pages in both servers can be downloaded up to the tolerance value. Because new servers pointed to undecided servers will also be treated as undecided, the crawling range can easily explode. On the other hand, decreasing the tolerance to limit the crawling range would raise the risk of misidentifying relevant servers as irrelevant (false negatives).

A possible solution to this problem could be to defer link extraction from pages in undecided servers until the servers are established as relevant. However, we adopt a simpler approach where the following step is executed after step 4 of Fig. 3.

4.5 When the page belongs to an undecided or irrelevant server [server i such that $N_r(i) = 0$], reject links to unvisited servers [all servers j such that $N_a(j) = 0$].

The new step limits the addition of new undecided servers to the neighborhood of relevant servers in order to avoid explosive growth of undecided servers. As for links



1. Identify language of the downloaded page.
2. If the page is relevant, increment $N_r(i)$, where i denotes the server to which the page belongs.
3. Increment $N_a(i)$.
4. Extract links from the page.
5. Reject links which point to irrelevant servers.
6. Reject links to already downloaded pages.
7. If the downloaded page is relevant, put the remaining links' destination URLs into the queue with a high priority.
8. Otherwise, put the remaining links' destination URLs into the queue with a low priority.

Fig. 3. Link selection processing for language-specific crawling.

to existing undecided servers [all servers i such that $0 < N_a(i) \leq T, N_r(i) = 0$], we follow them to establish the server's identification status as early as possible.

4.3. Use of finer granularity than servers

The proposed server-based identification method assumes intuitively that a Web server represents an individual or organization that has uniform nationality and mother tongue. However, server granularity may be too coarse, because there are also multilingual commercial sites and Web sites that host a variety of users. These sites tend to organize separate directory hierarchies for each language or user. Thus, we take into account the cases where language attributes are assigned to top-level directories (in terms of URL strings) of servers. That is, Fig. 3 is modified so that the number of downloaded pages $N'_a(i')$ of directory i' and the number of relevant pages $N'_r(i')$ replace $N_a(i)$ and $N_r(i)$, respectively. Examples of mapping from URLs to server directories are as follows:

- http://www.xsp.com/user1/whatsnew.html → www.xsp.com/user1/
- http://www.xsp.com/user2/diary/index.htm → www.xsp.com/user2/
- http://www.xsp.com/about.html → www.xsp.com/

This variation makes it possible to selectively acquire a server's specific directory hierarchies which contain relevant pages.

5. Evaluation

5.1. Crawling strategies tested

In this section, we present the result of the simulation-based evaluation of the crawling strategies proposed in Section 4 using the data set described in Section 2. The strategies used in our evaluations are:

Aggressive server-based filtering: allows any link to explore into new servers. This is our basic strategy described in Section 4.1.

Conservative server-based filtering: allows only links from relevant servers to explore into new servers (the strategy described in Section 4.2).

Conservative directory-based filtering: allows only links from relevant directories to explore into new directories (the strategy described in Section 4.3).

In addition, the following strategies are also evaluated for comparison.

Hard focused: discards links from irrelevant pages.

Soft focused: puts links from relevant pages into the queue with high priority, and those from irrelevant pages into the queue with low priority.

BFS: performs plain breadth-first search regardless of language attributes.

Perfect: follows only links that are known to lead to relevant pages. Note that the links that lead to relevant pages were determined in advance using breadth-first search on the data set.

We specified the same seed URLs as those used for collecting the data set. Although we performed tests with other seed URLs of Thai servers, the results remained almost the same.

5.2. Crawled pages

Figure 4 depicts the cumulative Thai page miss ratio versus the distance from seeds at the end of crawls with the proposed strategies and the hard focused strategy.† Note that the graph distances do not agree with the lengths of the actual crawl paths due to link rejection. As shown in Fig. 1, the data set does not contain enough pages beyond a distance of 7. Thus, miss ratios at large distances may lack accuracy.

According to Fig. 4, page misses occur from a distance of 3. The hard focused strategy shows the largest miss ratio, leaving nearly 10% of Thai pages uncrawled within a distance of 6. The next is the (conservative) directory-based filtering strategy, with which the miss ratio keeps increasing as the distance becomes larger. The strategy restricts the exploration of newly discovered directories to links from directories containing Thai pages. Because links originating from undecided directories were not reprocessed after they turned out to be relevant, pages in directories with few inlinks must have been missed. Hence, the directory-based filtering strategy is not suitable for archiving purposes where the loss of pages is unacceptable.

As for server-based filtering strategies, the miss ratios are not more than 1% and the curves are almost flat. Increasing the tolerance from 10 to 20 drops them slightly. With the aggressive server-based filtering strategy, the miss ratio approaches zero if we set the tolerance *T* large enough. Apparently, *T* must be greater than 5, at least 10. That is, server language identification requires that 10 irrelevant pages be tolerated. However, we can observe that misses still remain with even larger *T*, that is, 20. This means that some links from (already established) irrelevant servers are indispensable. As Fig. 2 shows, there are Thai pages reachable only through a sequence of several non-Thai pages. Thus, this is an essential limit of our strategies which depend on the language locality of pages and servers. Detailed analysis of individual cases is beyond the scope of this paper and is left for future work.

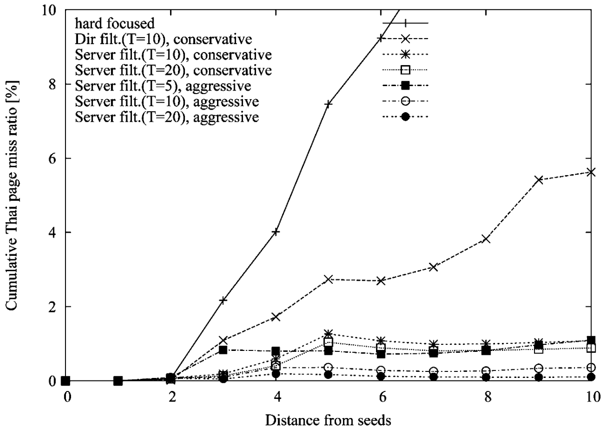


Fig. 4. Cumulative Thai page miss ratio at the end of crawls.

Figure 5 shows a plot of the coverage (recall) of Thai pages during the crawls. The mark on each curve shows the point where the crawling with the corresponding strategy halted. In BFS, the coverage increases gradually because the strategy makes no distinction between relevant and irrelevant pages. Contrarily, the curve denoted as "perfect" corresponds to an ideal selection of necessary and sufficient links, with perfect knowledge of the Web graph. The coverage curves of other strategies look similar and are much steeper than that of BFS. Since all of these strategies prefer links from Thai pages, they do not differ much so long as the Thai pages are abundant, which must be the case where the number of crawled pages is no more than 2 million. After 2 million pages have been crawled, the remaining Thai pages become scarce, so that the hard focused strategy halts while the changes of the soft focused search become gentle. On the other hand, server-based filtering strategies show steeper increases than the soft focused strategy and halt with most of the Thai pages covered. Although the aggressive strategy eventually reaches higher coverage (a lower miss ratio) than the conservative strategy, the former seems inappropriate due to its quite low efficiency at the final stage.

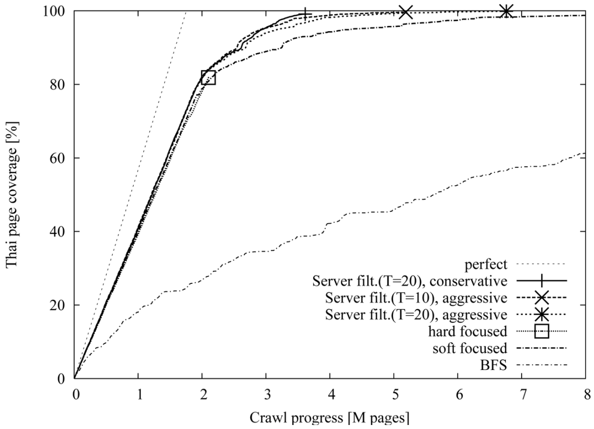


Fig. 5. Plot of Thai page coverage.

†The miss ratios of soft focused, BFS, and perfect strategies are 0, because they never discard links to Thai pages.

Figure 6 is a plot of the cumulative harvest rate (precision) of Thai pages (the ratio of the number of Thai pages crawled to the total number of crawled pages) during the crawls. As in Fig. 5, the marks indicate the termination points of the crawls. Until the number of crawled pages reaches 2 million, the harvest rate remains at about 70%, which coincides with the percentage of links between Thai pages in Table 4. Note that the server-based filtering strategies achieved a 2 to 3% higher harvest rate than the hard focused and soft focused strategies, which are page-based. This is due to the effect of rejecting links to non-Thai servers. We will need to reject more links from Thai pages to non-Thai pages to achieve even higher harvest rates.

As for the perfect strategy, the harvest rate gradually increases from 92% to around 99%. It is interesting that starting from a low harvest rate resulted in a higher overall harvest rate. This observation implies the effectiveness of a strategy which decreases the tolerance value as crawling proceeds.

5.3. Runtime efficiency

Here, we evaluate the proposed strategies in terms of the crawler's runtime efficiency, rather than the number of crawled pages. Figure 7 shows a plot of the queue size (the number of frontier URLs in the queue) during the crawls. As noted in Section 3, the Web graph derived from the data set contains uncrawled neighboring URLs. In Fig. 7, these URLs are assumed to stay in the queue once put there. Thus, at the right end of each curve, URLs with actual content in the data set are exhausted and only URLs with no content in the data set remain in the queue. In the soft focused and the aggressive server-based filtering strategies, the queue sizes drastically increase after 2 million pages have been downloaded, where the remaining Thai pages become sparse in the graph. Beyond this point, the crawler seems to step into the non-Thai Web. Hence, these strategies are inadequate to collect nationwide Web pages efficiently.

On the other hand, the queue size of the conservative server-based filtering strategy does not increase so much after collecting 2 million pages. Small tolerance values cause slightly earlier halts of crawling with similar curves. Though directory-based filtering and hard focused strategies show even smaller queue sizes, they have been revealed to be inappropriate in terms of coverage.

Figure 8 depicts a plot of the cumulative number of servers accessed during the crawls. Due to DNS (name resolution) overhead and resource consumption (e.g., sockets) associated with server access, it is undesirable to access too many servers. From this viewpoint, the conservative server-based filtering strategy shows the most appropriate behavior.

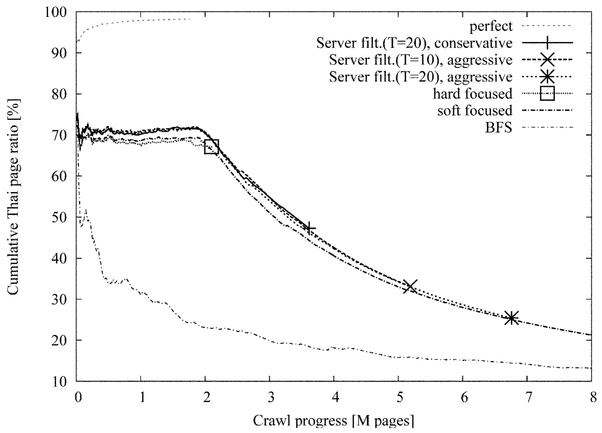


Fig. 6. Plot of cumulative Thai page harvest rate.

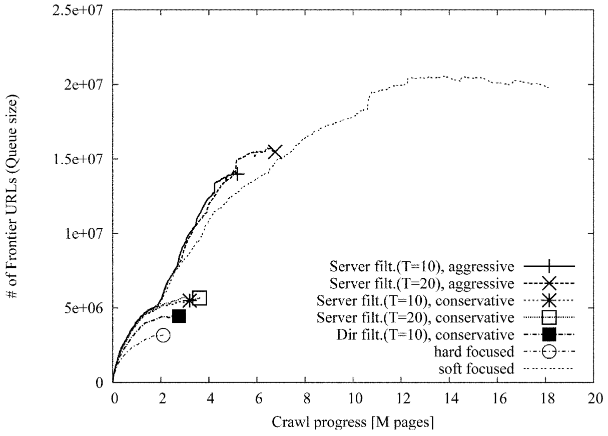


Fig. 7. Plot of crawler queue size.

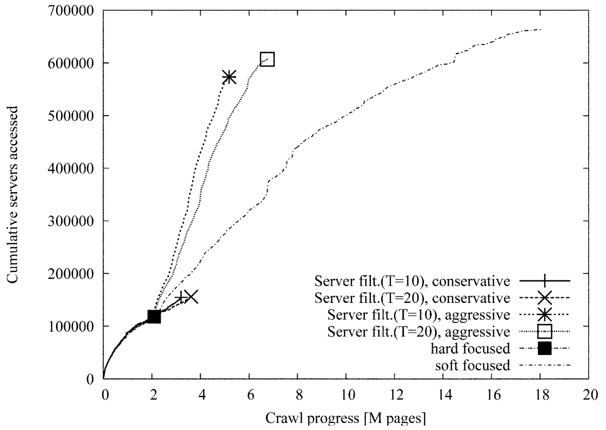


Fig. 8. Plot of cumulative number of servers accessed.

6. Conclusions

In this paper, we have proposed a language-specific Web crawling method as a measure for extracting information native to a nation or culture from the borderless Web. We evaluated the method using a Thai Web subset and confirmed its effectiveness. Because the scale of the nationwide Web crawling is much larger than that of the existing focused crawling methods, which deal with at most tens of thousands of Web pages, the performance requirements regarding both harvest rate and runtime efficiency are high. For the harvest rate, the language locality of the Web, where Web pages in the same language form a dense graph, enabled us to achieve good results (70% in the initial stage) relatively easily, by giving priority to the destinations of links in relevant pages. For languages that are more abundant on the Web, such as Japanese, the language locality will be more intensive. On the other hand, we found that runtime efficiency could easily drop without careful rejection of link destinations. Excess URLs could swell the crawler queue, thus raising the cost of its management, whereas excess servers could increase the delays involved in DNS queries and incur conflict over the network resources.

The method proposed in this paper makes it possible to immediately judge the relevance of newly discovered URLs by dynamically classifying relevant and irrelevant servers which represent the origins of Web pages. Furthermore, it also solves the problem of degradation of runtime efficiency by allowing only links from relevant servers to explore into newly discovered servers.

Concerning the purpose of preserving Web information, comprehensiveness of collection is crucial. The proposed method achieved about 99% page coverage. The essential cause of the few misses must be the existence of Web pages linked only via non-Thai pages/servers. We discovered a Thai page which could not reach another Thai page without passing through at least 8 non-Thai pages. In future work, we will analyze such instances in more depth to seek an even better strategy to improve coverage and harvest rate.

Acknowledgment. Part of this research was supported by the Comprehensive Development of e-Society Foundation Software program of the Ministry of Education, Culture, Sports, Science and Technology of Japan.

REFERENCES

1. International Internet Preservation Consortium. netpreserve.org. http://netpreserve.org/.
2. National Diet Library Japan. WARP: Web ARchiving Project. http://warp.ndl.go.jp/.
3. Internet Archive. About IA. http://www.archive.org/about/about.php.
4. Chakrabarti S, van den Berg M, Dom B. Focused crawling: A new approach to topic-specific Web resource discovery. Proc WWW8, 1999.
5. Chakrabarti S, Punera K, Subramanyam M. Accelerated focused crawling through online relevance feedback. WWW '02: Proceedings of the 11th International Conference on World Wide Web, New York. ACM Press; 2002. p 148–159.
6. Diligenti M, Coetzee F, Lawrence S, Giles CL, Gori M. Focused crawling using context graphs. 26th International Conference on Very Large Databases, VLDB 2000, Cairo, p 527–534.
7. Cavnar WB, Trenkle JM. N-gram-based text categorization. Proceedings of SDAIR-94, 3rd Annual Symposium on Document Analysis and Information Retrieval, Las Vegas, p 161–175, 1994.
8. Suzuki I, Mikami Y, Ohsato A, Chubachi Y. A language and character set determination method based on n-gram statistics. ACM Transactions on Asian Language Information Processing (TALIP) 2002;1:269–278.
9. The Mozilla Organization. Mozilla project page (1998). http://www.mozilla.org/.
10. Li S, Momoi K. A composite approach to language/encoding detection. Proc 19th International Unicode Conference, 2001.
11. Baeza-Yates R, Castillo C, Marin M, Rodriguez A. Crawling a country: Better strategies than breadth-first for Web page ordering. WWW '05: Special interest tracks and posters of the 14th International Conference on World Wide Web, New York. ACM Press; 2005. p 864–872.
12. Ghani R, Jones R, Mladenic D. Mining the Web to create minority language corpora. CIKM; 2001. p 279–286.
13. Internet Assigned Numbers Authority. Character sets (2005). http://www.iana.org/assignments/character-sets.
14. WiseGuys Internet B.V. libTextCat—Lightweight text categorization (2003). http://software.wise-guys.nl/libtextcat/.

AUTHORS (from left to right)

Takayuki Tamura received a B.E. degree in electronic engineering and M.E. and Ph.D. degrees in information engineering from the University of Tokyo in 1991, 1993, and 1998. He is currently at Mitsubishi Electric Corporation and has been engaged in joint research with the Institute of Industrial Science of the University of Tokyo. His research interests include Web crawling, Web mining, and parallel database processing. He is a member of ACM, the IEEE Computer Society, IEICE, the Information Processing Society of Japan, and the Database Society of Japan.

Kulwadee Somboonviwat is a Ph.D. candidate in the Information and Communication Engineering Department of the University of Tokyo. Her research interests include Web crawling and Web archiving. She received her M.E. degree in information science from the University of Tokyo in 2005, and her B.E. degree in computer engineering from King Mongkut's Institute of Technology, Ladkrabang, Thailand, in 2000.

Masaru Kitsuregawa received a Ph.D. degree from the University of Tokyo in 1983. He is currently a professor and a director of the Center for Information Fusion at the Institute of Industrial Science of the University of Tokyo. His current research interests include database engineering, Web mining, parallel computer architecture, parallel database processing/data mining, storage system architecture, digital earth, and transaction processing. He had been a VLDB Trustee and served as the general chair of ICDE 2005 at Tokyo. He is currently an Asian coordinator of the IEEE Technical Committee on Data Engineering, and a steering committee member of PAKDD and WAIM. In Japan, he chaired the data engineering technical group of IEICE, and served as ACM SIGMOD Japan Chapter Chair. He is currently an adviser to the Storage Networking Industry Association Japan and a director of the Database Society of Japan. He is a member of ACM and the IEEE Computer Society, and a fellow of IEICE and the Information Processing Society of Japan.