Quick Acquisition of Topic-based Information/Knowledge from News Site Databases

Hao Han
National Institute of Informatics (NII), Japan
han@nii.ac.jp

Abstract—Web news is an important resource of information/knowledge. We can analyze news to observe the difference in various topics (e.g., economy, health, and culture) and trends in the past years. However, the collection of topic-based Web news is considered as a long-period process usually. In this paper, an effective and efficient Web-based knowledge acquisition approach is proposed to extract topic-based Web news full contents from the news site databases directly. This approach is applicable to the general news sites, and the experimental results show that it can extract the topic-based Web information/knowledge from news site databases automatically, quickly and accurately.

Index Terms—Web-Based Tools, Knowledge Acquisition, News Extraction, Database

I. INTRODUCTION

Nowadays, fresh news contents on a variety of topics are being created and made available on the Web at breathtaking speed. We can analyze them to acquire the desired information/knowledge. For example, if we want to compare the monthly topics of each country in the past years from CNN, we need to collect the CNN news about each country and analyze these news contents to learn the desired information for personal use (not anti-copyright republication).

However, the process of the news pages collection consumes much time. Usually, the Web pages crawlers are used to collect the Web pages. They are executed at regular intervals, and the collection process has to last for a long period of time if we want to collect the news pages of long period. We do not think each one of the collected Web pages is usable because there are many non-news Web pages, such as the blog pages, advertisement pages and even similar pages with different URLs. Sometimes, we just want to collect the news with specific topics such as the news on “soccer” or “whaling”, and the other collected news are undesired. Furthermore, the news sites are crawled to find as many news pages as possible, but actually, it is difficult to acquire the old news pages because the latest news are shown prior to the old news. Besides, in a news page, there are advertisement, related stories and other undesired parts usually. In order to recognize and extract the parts of news contents from the news pages, the extraction patterns are generated based on the layout of news pages. Web page layout is the style of graphic design in which text or pictures are set out on a Web page. The different news sites use the different news pages layout, and each news site uses more than one layout usually. It is necessary to generate many news contents extraction patterns manually or automatically for each news site. It is a costly work. Moreover, the news sites update the layout of news pages irregularly. If the news sites update the layout of news pages, the corresponding analysis has to be done again. Therefore, it is not easy to extract news contents on the specific topics from news sites quickly, and the current methods of news pages collection and news contents extraction can not work efficiently.

In this paper, we propose an approach to extract the topic-based Web information/knowledge from news site databases quickly. Usually, the news sites provide site-side news search engines for the users. These engines are affiliated to the news sites and can access the news databases of news sites directly. We use these news search engines to search for the news by giving the keywords of specific topics, and extract the page URLs and titles of the matched news from the search result pages automatically. Then we use an efficient extraction algorithm to extract the full contents of the news without Web page layout analysis. We can designate the target news sites, publication dates/periods (e.g., last week, this month, from 2008 to 2010) and topics. A topic is a discrete piece of content that is about a subject, such as a series of countries, sports, companies and etc. Our approach is applicable to the general news sites and can extract a large number of news including the old ones published some years ago. Our main purpose is to provide a practical and easy-to-use Web-based information/knowledge acquisition tool for news-oriented research.

The organization of the rest of this paper is as follows. In Section 2 we give the motivation of our research and an overview of the related work. In Section 3, 4 and 5, we explain our topic-based Web information/knowledge extraction approach in detail. We test our approach and give an evaluation in Section 6. Finally, we conclude our approach and give the future work in Section 7.

II. MOTIVATION AND RELATED WORK

In order to realize the analysis and comparison of Web news in many major topics, it is necessary to collect the news with specific topics from one or many designated news sites over a long period of time. Some related work has been done on Web news collection or extraction. For the news pages collection, the Web pages crawlers are often used. They are executed to collect the news pages from news sites and the collection process costs much time. Several collection approaches and systems have been proposed. More and more news sites distribute news by RSS. Generally, news sites classify the
news into different categories and publish them by RSS feeds. However, different news site uses different categories and RSS feeds just comprise the latest news. For example, CNN provides RSS feeds by fields such as science, sports, business and etc, while AllAfrica (allAfrica.com) offers RSS feeds grouped by countries/regions. AllInOneNews (www.allinonenews.com) is a news search system based on automatic extraction of search results from search engines [9]. It passes each user query to the existing search engines of news sites, collects their search results for presentation to the user. However, the users of this system can not select the target news sites, and just collect the results from the first search result page. Google News (news.google.com) provides the news search service and distributes the news search results by RSS or Atom. If we use the default or advanced search of Google News, we can select the target news sites, but the publication date/period selection is weak. If we use the archive search, we can not select the target news sites. If we use the search result RSS feeds, only the results from the first search result page can be collected. These methods/systems can not satisfy the flexible and quick collection of news pages very well. Moreover, these methods can not realize the comprehensive analysis or comparison of news because they can not extract the full contents of each news. They can not easily answer the questions like "which countries had an argument over whaling during the last years and whether the other countries were attracted to discuss it as the arguments went on".

For the news contents extraction, a number of approaches have been proposed to analyze the layout of the news pages with the purpose of manual or semi-automatic example-based information extraction pattern learning, and to extract the news contents from the general news pages ultimately. Reis et al. gave a calculation of the edit distance between two given trees for the automatic Web news contents extraction [2]. Fukumoto et al. gave the focus on subject shift and presented a method for extracting key paragraphs from documents that discuss the same event [3]. It uses the results of event tracking which starts from a few sample documents and finds all subsequent documents. However, if a news site uses too many different layout in the news pages, the learning procedure costs too much time and the precision becomes low. Zheng et al. presented a news page as a visual block tree and derived a composite visual feature set by extracting a series of visual features, then generated the wrapper for a news site by machine learning [10]. However, it uses manually labeled data for training and the extraction result may be inaccurate if the training set is not large enough. Webstemer [7] is a Web crawler that automatically extracts main text of a news site without having banners, advertisements and navigation links mixed up. It analyzes the layout of each page in a certain web site and figures out where the main text is located. All the analysis can be done automatically with little human intervention. However, this approach runs slowly at contents parsing and extraction, and sometimes news titles are missing. TSRec [6] provides a hybrid method for news contents extraction. It uses tag sequence and tree matching to detect the parts of news contents from a target news site. However, for these methods, if the news sites change the layout of news pages, the analysis of layout or tag sequence has to be done again. As the layout-independent extraction approaches, TidyRead (www.tidyread.com) and Readability (lab.berkelely.edu/measure) render Web pages with better readability as an-easy-to-read manner by extracting the context text and removing the cluttered materials. They run as plug-in or bookmarklet of Web browser. However, the extraction result is a part of Web page containing the HTML tags. It also contains some other non-news elements such as the related links. Wang et al. proposed a wrapper to realize the news extraction by using a very small number of training pages based on machine learning processes from news sites [8]. The algorithm is based on the calculation of the rectangle sizes and word numbers of news title and contents. However, these approaches still need to set the values of some parameters manually, and could not be proved to extract the news successfully or automatically if news sites update the page layouts. Full-Text RSS (echodittolabs.org/fulltextrss) only returns the news contents when the supplied RSS has a summary or description of some kind.

These news contents extraction methods are still not widely used, mostly because of the need for high human intervention or maintenance, and the low quality of the extraction results. Most of them have to analyze the news pages from a news site before they extract the news contents from this news site. If we select the different target news sites, topics and publication dates, the analysis of layout needs to be done again. It is costly and inefficient. Compared to these developed work, we use the news search engines affiliated to the news sites instead of the often used Web crawlers. We can get a large number of news from the news site databases, not only the latest news but also the old news. The target news sites, topics and publication dates are selective. Furthermore, we do not need to delete the non-news pages or other undesired news pages from the search results because all the news extracted from search result pages satisfy our designated topics. Meanwhile, we propose an algorithm special for the news contents extraction. It is applicable to the general news pages, and we do not need to analyze different kinds of news pages to generate the corresponding extraction patterns for each news site. The full contents of news are quickly extracted from the matched news pages for the further analysis.

III. OVERVIEW

Our approach is made up of two parts mainly as shown in Fig. 1: news pages collection and news full contents extraction. Firstly, we collect the topic-based news pages. We create a submitting emulator to emulate the submitting process of search engine of the target news site. We get the search keywords of a specific topic and send them to the emulator one by one, then extract news titles and URLs of news pages from the continuous search result pages. Secondly, we extract the news contents from the news pages. We propose an extraction
algorithm special for news pages, which can extract the news contents from a news page only by using the news title.

![Fig. 1. Overview of our system](image1)

### IV. NEWS PAGES COLLECTION

We collect news pages from news site search engines. Although many Web sites, such as Amazon and YouTube, open their search engine services by Web service APIs, most of the news sites, such as CNN and BBC, do not provide Web services for their news search engines. We have to extract the partial information, such as news titles, news page URLs and publication dates, from the news search result pages. As shown in Fig. 2, we generate a submitting emulator for designated news site, and send the search keywords to the submitting emulator to receive the search result pages. Then, we analyze the search result page to extract the news titles and URLs.

![Fig. 2. Overview of news pages collection](image2)

#### A. Submitting Emulator

Usually, in a news Web site, there is a site-side search engine used to get the requests from users and return the search result pages. The users enter the query keywords into a form-input field by keyboard and click the submit button by mouse to send the query. For the request submitting, there are POST method and GET method, and some news Web sites use the encrypted codes or randomly generated codes. In order to get the search result pages from all kinds of news sites automatically, we use HtmlUnit (htmlunit.sourceforge.net) to emulate the submitting operation instead of URL templating mechanism.

We need to get the start Web page which comprises the form-input field and submit button of search engine. Usually, this start Web page is the top page of news site or a search result page of news site. Then we analyze the HTML document of this Web page to find the `<form>` nodes. If a form comprises a text input field and a button next to this text input field, and the server-side form handler of this form is within this news site, we think it is a possible form which includes the necessary form-input field and submit button. If we find more than one possible form in this Web page, we choose the first one as our final selection because the target form is at the top side of Web page usually.

We generate a submitting emulator and send the search keyword to it. Submitting emulator uses HtmlUnit to emulate the submitting process (input the search keyword into text field and click the button to complete the actual submit). Finally, we get the response page (search result page) from submitting emulator. All the processes of submitting emulator generation in our approach are completed automatically after the start Web page is given.

#### B. News Title and News Page URL Extraction

After we get the search result page, we need to extract the news titles and news page URLs. There are links to the advertisement pages, video pages, external non-news pages and other irrelevant information besides the page links to matched news. Fortunately, there are some similar features in the news search result pages of most news sites, which can be used to extract the news titles and links to the news pages.

- Each search result set contains the similar information at the similar position such as news page link, news title, headline and publication date.
- The news title is contained inside the news page link as text value.
- Matched news are listed in a column and spread over multi-pages.
- All the news search result pages from a news site search engine have the same Web page layout.

We extract all the link nodes from the HTML document of search result page, and find out the news page link nodes. Through our analysis of search result pages of many news sites, we find that the news page link node has some common features in its text and path (XPath expression). We calculate the possibility value of link node by the following steps. Usually, a larger possibility value represents the corresponding link node is more possible to be a news page link node.

1. We split the text value of link node into word list \( WL \) using whitespace as the delimiter, and get the length of \( WL \) as \( L_1 \) \((L_1 \geq 1)\).
2. We calculate the occurrence time of search keyword in \( WL \) as \( L_2 \) \((L_2 \geq 0)\).
3. We get the path of link node including the ID and class value. We calculate the occurrence time of "news" and "search" and "result" in path respectively. We count these three values up to get the sum value \( L_3 \) \((L_3 \geq 0)\).
4. We calculate the possibility value of each link as \( P \) by using the following formula.
where, $\alpha = 0$ if $L_2 > 0$, and $\alpha = 1$ if $L_2 = 0$. Similarly, $\beta = 0$ if $L_3 > 0$, and $\beta = 1$ if $L_3 = 0$. We can not make certain that the search keyword must occur in the news title because the search range contains not only the news title but also the news contents, which is not visible in the search result page. Also, the value of $L_3$ may be 0 in many news sites. We use $\alpha$ and $\beta$ to avoid the possible occurrence of $P=0$. They work well and do not bring the negatives to the actual extraction in our experiments.

In some news search result pages, the link node with the largest possibility value is not a link node of the news page always (e.g., a link to contextual advertising or blog). Usually, the news page links are listed in the similar structures (XPath) and not mixed with other non-news links. We use the following steps to detect the news page link nodes range and find out a news page link node.

1) We count up the possibility value $P_N$ of all the link nodes, and get the root mean square $RP$ as a threshold.

$$RP = \sqrt{\frac{\sum P_N^2}{|N|}}$$

where, $|N|$ is the sum of link nodes.

2) For each node $N$, we calculate $P_N$ as follows.

$$P_N = \begin{cases} 
P_N & (N \text{ is a link node}) \\
\sum_{n \in \text{Child}_N} P_n & (P_n > RP)
\end{cases}$$

where, $\text{Child}_N$ is a set of child nodes of the node $N$.

Fig. 3 shows an example of calculation of $P$.

3) We select the child node whose value is the largest among the sibling nodes from root node to leaf nodes as shown in Fig. 4.

We think the final selected child node is the link node of a news page, and use the path of this node to extract the list of nodes with the similar paths (like WIKE [5]). Each node of list represents a news page link node, and the text value from each node is the news title.

If the paths of these nodes show that these nodes are listed in a column, we think they are the final extraction results and represent the news page links because most news sites show the search results in a column, not in a row. Otherwise, we think the search result page does not comprise the matched news and shows the message like "No Results Found" because the search engine does not find the corresponding news about the search keyword in news database.

The news search results are spread over multi-pages and we need to extract the page number links for our continuous query and extraction. The extraction of these links has to satisfy the following rules.

1) The text values of links are a series of numbers such as 1, 2, 3, ....
2) The href attribute values of links have the similar length.
3) The links are listed in a row.

We use the paths of news page link nodes to extract the news page link nodes directly from the second result page. If we search for the news for many search keywords continually in a news site, we use these paths to extract the news page links and page number links directly.

C. Publication Date Extraction

The publication date is necessary if we collect the news of a specific period. For example, we need to extract the news of baseball of the last 5 years if we want to find out which team was the annual focus of attention in the last 5 years.

Different news sites choose the different formats of date information such as "March 7, 2011", "Mar. 7, 11" and "2011-3-7". Table I shows the most used format patterns of date information. We use these patterns to find the publication date in search results. Usually, a news site displays the publication date at the same position in a search result and uses a same pattern for all the publication dates in all search results. After we find a publication date in a search result, we can use the similar patterns and same pattern to extract other news publication dates from search results easily.

V. News Full Contents Extraction

We extract the full contents of news from our collected news pages. The news pages from different news sites use the different page layout and news sites update their news page
layout irregularly. We propose our news contents extraction algorithm, which is independent of the layout of news page and applicable to the general news pages. We detect the position of a news title in the news page and extract the body of news (paragraphs of news contents).

The first process detects position of a news title in the obtained news page. The news title is a piece of important information for the recognition of the news contents from the full text of news page. If we correctly locate the position of the title in a news page, the position of news contents text would be found easily because the contents text is a list of paragraphs closely preceded by the title usually. In addition, for a news, the contents describe the same topic of news title in detail, and the words constituting the title would occur in the news contents frequently usually.

The second process detects a part of the news body and extracts the whole body. Since body of a news is usually preceded by its title, the process tries to find the body in range, it tries to find the body in a "reserve range", "Contents range" and "reserve range" are parts which might include the news body. We gave a detailed description of extraction algorithm in [4].

VI. EVALUATION

In this section, we give the experiments to test our algorithms and analyze the experimental results to evaluate our approach. We use the news sites listed in Table II as our test bed. These news sites are the popular on-line news publishers, including the global and domestic news sites.

A. Experiment 1

We selected the countries/regions and their leaders as our test topics. There are 242 countries/regions in the world and most of them have the leaders [1]. We used these country/region names and leader names as our search keywords. We collected the news page URLs and titles from the 10 (No.1-No.10, if the total number of pages is less than 10, we got them all) result pages of each keyword. As shown in Table II, Page Collection is the average execution time of extracting the news page URLs and titles from one result page (including the submitting emulation process), and Contents Extraction is the average execution time of extracting the news contents from one news page. The submitting emulation of two news sites (Washington Post and Xinhuanet) failed, and the corresponding Contents Extraction values are calculated by extracting news contents from manually collected/saved result pages. We selected 500 news page URLs randomly and checked them one by one manually, and found that 17 news pages could not be obtained (the server responded the message like "page not found").

B. Experiment 2 (precision ≃ 97.0%)

We sent the keywords used in Experiment 1 to submitting emulator one by one, and extracted 96,095 news titles and page URLs of matched news (published from January 1, 2003 to December 31, 2007) from news database of CNN. Our computer (CPU: Intel Pentium M 1.30GHz, Memory: 1.0GB RAM, Network: 54.0Mbps Wireless) uses about 20 hours to complete this extraction process. We selected 200 news page URLs randomly and checked them one by one manually. The experiment results are listed in Table III. We found that 2 news pages could not be obtained as the reasons described in Experiment 1. Among the rest 198 news pages, the news article contents of 192 news pages are extracted correctly. In the 6 extraction failures, some parts of news article contents are not extracted.

C. Experiment 3 (precision ≃ 97.4%)

We have crawled and extracted more than 1.8 million news contents from 38 famous news sites [4] since 2007. We select 2500 news articles randomly and check them one by one manually. The experiment result is listed in Table IV.

D. Analysis and Evaluation

We use the first and second experiments to test our submitting emulator and news page collection algorithm. It proves that our approach can extract the news titles and URLs of news pages of a long period from news site databases easily and

<table>
<thead>
<tr>
<th>Pattern</th>
<th>Component</th>
<th>Format</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>YY</td>
<td>Year</td>
<td>2-digit number</td>
<td>11</td>
</tr>
<tr>
<td>YYYY</td>
<td>Year</td>
<td>4-digit number</td>
<td>2011</td>
</tr>
<tr>
<td>MMM</td>
<td>Month</td>
<td>Text</td>
<td>Mar, March</td>
</tr>
<tr>
<td>MM</td>
<td>Month</td>
<td>Number</td>
<td>3, 03</td>
</tr>
<tr>
<td>DD</td>
<td>Day</td>
<td>Number</td>
<td>7, 07</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Country/Region</th>
<th>News Site</th>
<th>Page Collection (second)</th>
<th>Contents Extraction (millisecond)</th>
</tr>
</thead>
<tbody>
<tr>
<td>United States</td>
<td>CNN</td>
<td>14.3</td>
<td>6.02</td>
</tr>
<tr>
<td></td>
<td>New York Times</td>
<td>4.6</td>
<td>0.63</td>
</tr>
<tr>
<td></td>
<td>Washington Post</td>
<td>×</td>
<td>(6.62)</td>
</tr>
<tr>
<td>United Kingdom</td>
<td>BBC</td>
<td>3.6</td>
<td>3.19</td>
</tr>
<tr>
<td>Africa</td>
<td>All Africa</td>
<td>7.3</td>
<td>2.98</td>
</tr>
<tr>
<td>China</td>
<td>Xinhuanet</td>
<td>×</td>
<td>(2.94)</td>
</tr>
<tr>
<td>France</td>
<td>France 24</td>
<td>8.6</td>
<td>3.07</td>
</tr>
<tr>
<td>Japan</td>
<td>Mainichi Daily News</td>
<td>9.3</td>
<td>2.80</td>
</tr>
<tr>
<td>South Korea</td>
<td>Chosun Ilbo</td>
<td>6.6</td>
<td>0.32</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Country/Region</th>
<th>News Site</th>
<th>Page Collection (second)</th>
<th>Contents Extraction (millisecond)</th>
</tr>
</thead>
<tbody>
<tr>
<td>United States</td>
<td>CNN</td>
<td>14.3</td>
<td>6.02</td>
</tr>
<tr>
<td></td>
<td>New York Times</td>
<td>4.6</td>
<td>0.63</td>
</tr>
<tr>
<td></td>
<td>Washington Post</td>
<td>×</td>
<td>(6.62)</td>
</tr>
<tr>
<td>United Kingdom</td>
<td>BBC</td>
<td>3.6</td>
<td>3.19</td>
</tr>
<tr>
<td>Africa</td>
<td>All Africa</td>
<td>7.3</td>
<td>2.98</td>
</tr>
<tr>
<td>China</td>
<td>Xinhuanet</td>
<td>×</td>
<td>(2.94)</td>
</tr>
<tr>
<td>France</td>
<td>France 24</td>
<td>8.6</td>
<td>3.07</td>
</tr>
<tr>
<td>Japan</td>
<td>Mainichi Daily News</td>
<td>9.3</td>
<td>2.80</td>
</tr>
<tr>
<td>South Korea</td>
<td>Chosun Ilbo</td>
<td>6.6</td>
<td>0.32</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Country/Region</th>
<th>News Site</th>
<th>Page Collection (second)</th>
<th>Contents Extraction (millisecond)</th>
</tr>
</thead>
<tbody>
<tr>
<td>United States</td>
<td>CNN</td>
<td>14.3</td>
<td>6.02</td>
</tr>
<tr>
<td></td>
<td>New York Times</td>
<td>4.6</td>
<td>0.63</td>
</tr>
<tr>
<td></td>
<td>Washington Post</td>
<td>×</td>
<td>(6.62)</td>
</tr>
<tr>
<td>United Kingdom</td>
<td>BBC</td>
<td>3.6</td>
<td>3.19</td>
</tr>
<tr>
<td>Africa</td>
<td>All Africa</td>
<td>7.3</td>
<td>2.98</td>
</tr>
<tr>
<td>China</td>
<td>Xinhuanet</td>
<td>×</td>
<td>(2.94)</td>
</tr>
<tr>
<td>France</td>
<td>France 24</td>
<td>8.6</td>
<td>3.07</td>
</tr>
<tr>
<td>Japan</td>
<td>Mainichi Daily News</td>
<td>9.3</td>
<td>2.80</td>
</tr>
<tr>
<td>South Korea</td>
<td>Chosun Ilbo</td>
<td>6.6</td>
<td>0.32</td>
</tr>
</tbody>
</table>
quickly. Our approach is applicable to the general news site search engines and does not need the methods like machine learning or extraction pattern matching, which cost much time when news sites change the layout of search result pages. However, some news sites use the external JavaScript files comprising the complicated JavaScript functions to realize the request submitting, or even the minor syntax errors occur in Web pages where the search keywords are inputted. Although the most of the current Web browsers, such as Firefox and Internet Explorer, can run smoothly on these Web pages, our submitting emulator still can not emulate this kind of submitting processes. We think it is a bug of HtmlUnit and wish the new version would solve this problem in the future. Furthermore, the emulation processes of some news sites run slowly. For example, the emulation of submitting search keyword to CNN news search engine costs about 10 seconds. Moreover, some old news pages are not obtained though their URLs and news titles are shown in news search result pages.

We use the second and third experiments to test our news full contents extraction algorithm. We use a large number of news to test our algorithm and the experimental results prove that our extraction algorithm is highly accurate over a long period of time. Although the news sites change the layout of news pages irregularly, our extraction method works well and the precision of extraction is over 97%. However, in some news pages, a paragraph, usually the outline of news, shows in different style compared to other paragraphs. This kind of paragraph looks like a non-news part such as an advertisement in text format, and is omitted in the extraction. Moreover, some news contents are too short to recognize from the news pages. For example, a news flash about baseball game result, which contains just a short paragraph of ten words, maybe can not be extracted correctly.

Compared with other developed extraction systems, our extraction approach has the following strong points.

1) Our extraction system is constructed easily, even for the users who know little about the information extraction technologies. The extraction processes run automatically, such as submitting emulator generation, news pages collection and news contents extraction. It needs little maintenance during the long period extraction. We do not need to analyze the layout of search result pages and news pages of news sites since our extraction algorithm is independent of the layout of Web pages. It does not need to reconfigure extraction even though the news sites change the layout of news pages.

2) Our extraction system supports the designation of news collection/extraction range, such as the target news site, news topic and publication date. By analysis of extracted news contents, we can compare the viewpoint of a topic among different news sites, see monthly/yearly variation of a topic, observe co-occurrence of one or two country names, and find other useful information/knowledge.

3) Our extraction system runs quickly because of simple and efficient algorithms. For the extraction of a large number of news, a simple algorithm of low computational complexity saves a considerable amount of time. For example, the contents extraction from a CNN news page costs 6.02 milliseconds averagely (excluding reading news pages from news sites and saving the extraction results into local hard disk), which is more efficient than other developed methods.

VII. Conclusion

In this paper, we have presented an effective and efficient approach to realize the quick and automatic extraction of topic-based Web information/knowledge from news site databases by using the site-side search engines. We proposed an algorithm to extract the news titles and news page URLs from search result pages. We also proposed an algorithm to extract the news full contents from news pages. Our extraction methods are applicable to the general news sites. All the processes of extraction are completed automatically. Our experimental results on several news sites show that our extraction system works well and the proposed approach is very promising.

As future work, we will modify our algorithm to improve the accuracy rate even further, and observe difference in various topics among countries/regions to discover useful information and knowledge from news sites. Moreover, we will extend our approach to different kinds of information/knowledge sites and construct the corresponding analysis system.

VIII. Acknowledgement

We gratefully acknowledge the advice and support from Bin Liu (Yahoo! Japan), Takehiro Tokuda (Tokyo Tech) and Keiizo Oyama (NII). This work was partially supported by a Grant-in-Aid for Scientific Research A (No.22240007) from the Japan Society for the Promotion of Science (JSPS).

REFERENCES