WIKE: A Web Information/Knowledge Extraction System for Web Service Generation

Hao Han and Takehiro Tokuda
Department of Computer Science, Tokyo Institute of Technology
Meguro, Tokyo 152-8552, Japan
{han, tokuda}@tt.cs.titech.ac.jp

Abstract

We have a tremendous amount of information/knowledge available on the Web today. We often have a situation in which we need to collect partial contents of a whole page from one or a number of Web sites. Examples are collection of capital city names and population data from country profile sites or collection of company names and their industrial fields from finance sites. We present WIKE, a system for partial information extraction from Web pages without programming. We also give its applications to Web service generation.

Keywords: information extraction, Web service

1 Introduction

Today, we have a tremendous amount of information/knowledge available on the Web today. We often have a situation in which we need to collect partial contents of a whole page from one or a number of Web sites. For example, in the BBC Country Profiles [1], there exists a collection of 200 or more country/region information such as capital city, population and latest leader’s information. One Web page only has the information of a single country/region. If we would like to collect the information of all the countries/regions, the use of Web browsers would be a time-consuming tedious task. Similar tasks may be collection of disease names and corresponding parts of human body from health/medicine sites or collection of company names and corresponding industrial fields from finance sites.

We present a demo of our system, called WIKE, for Web Information/Knowledge Extraction from the general Web applications without programming. The users can pick up the target Web pages, select the desired parts, and extract them by using WIKE. The extraction result is structured data in XML format.

The purpose of this demo is to extract Web information/knowledge from Web sites and generate Web services quickly. Web services are becoming a standard method of sharing information/knowledge, and integrations of Web services are very useful. Unfortunately, many Web sites do not provide the Web services. Web applications are still the main methods for Web information/knowledge sharing.

In this demo, we show how to select the target information from the Web applications, how to compose the structured data, how to configure the Web services, and how to complete all these processes quickly via WIKE.

2 Web Information/Knowledge Extraction

We give an overview of WIKE as shown in Fig. 1. WIKE is based on XML tree approach. Firstly, we get a typical Web page from the target Web application, and define the names and data types for the selected parts to generate an extraction pattern. Then, WIKE uses the extraction pattern to extract the partial information from the ordinary Web pages, and returns a resulting table to show the structured extraction result.

Figure 1. Outline of WIKE

In this section, we demonstrate how to use WIKE to ex-
tract the country information about the listed 200 or more countries/regions from BBC Country Profiles site. The country information includes country name, country map, country facts (capital city, population, and etc.), leader’s photo and leader’s information.

2.1 Parts Selection and Data Types Definition

Parts selection represents "Which parts of page are needed?", and data type represents "What kind of information is needed?". The data type includes two kinds of information: property and structure. Property is text, object, or link. Text is the character string in Web pages such as an article. Object is one instance of the photo, video and other multimedia file. Link is a reference in a hypertext document to another document or other resource. Structure is single occurrence or continuous occurrence. A single occurrence is a node without similar sibling nodes such as the title of an article. A continuous occurrence is a list of nodes with similar paths such as result list in a search result page. There are six kinds of data types: single text, continuous text, single object, continuous object, single link and continuous link.

2.2 Information Extraction Method

WIKE uses the defined data types and paths to realize the partial information extraction from the ordinary country pages. In the tree structure of HTML document, each path represents a node. WIKE extracts the nodes according to the corresponding paths. During the node extraction, if a node can not be found by a path, a similar path would be used to try extracting the node. We give a definition of similar path.

Similar Path: Two paths are similar to each other, if these two paths have the same forms ignoring the difference of orders of nodes among sibling nodes, and the difference of orders is within a given deviation range. The form of path is as follows:

\[
\text{body} : 0 : ID/N_1 : (O_1 - h \sim O_1 + h) : ID_1/N_2 : (O_2 - h \sim O_2 + h) : ID_2/.../N_{n-1} : (O_{n-1} - h \sim O_{n-1} + h) : ID_{n-1}/N_n : (O_n - h \sim O_n + h) : ID_n
\]

where, \( N_n \) is the name of the n-th node, \( O_n \) is the order of the n-th node among the sibling nodes, and \( N_{n-1} \) is the parent node of \( N_n \).

2.3 Extraction Pattern

In BBC Country Profiles, most of the country pages are similar to each other. The paths of the similar parts of these country pages are similar to each other, too. However, there

![Figure 2. Parts Selection and Data Types Definition](image)

Fig. 2 shows the parts selection and data types definition. We enter the URL of top page of BBC Country Profiles. WIKE lists the URLs of country pages. We select a typical country page. We define the names and data types for each selected part: country name part is CountryName of single text type, country map part is CountryMap of single object type, country facts part is Facts of continuous text type, leader’s photo part is LeaderPicture of single object type, and leader’s information part is LeaderInfo of single text type. Each selected part is represented by a node, and each node can be represented by its path from the root. We use the following form to save the path of a selected part:

\[
\text{Table 1. Partial Information Extraction}
\]

<table>
<thead>
<tr>
<th>Data Type</th>
<th>Partial Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>single text</td>
<td>node value of single leaf node</td>
</tr>
<tr>
<td>single object</td>
<td>attribute value of single node</td>
</tr>
<tr>
<td>single link</td>
<td>embedded link value of single node</td>
</tr>
<tr>
<td>continuous text</td>
<td>leaf node values of list of nodes</td>
</tr>
<tr>
<td>continuous object</td>
<td>attribute values of list of nodes</td>
</tr>
<tr>
<td>continuous link</td>
<td>link values of the list of nodes</td>
</tr>
</tbody>
</table>

"Which parts of page are needed?" and data type represents "What kind of information is needed?". The data type includes two kinds of information: property and structure. Property is text, object, or link. Text is the character string in Web pages such as an article. Object is one instance of the photo, video and other multimedia file. Link is a reference in a hypertext document to another document or other resource. Structure is single occurrence or continuous occurrence. A single occurrence is a node without similar sibling nodes such as the title of an article. A continuous occurrence is a list of nodes with similar paths such as result list in a search result page. There are six kinds of data types: single text, continuous text, single object, continuous object, single link and continuous link.

![Figure 2. Parts Selection and Data Types Definition](image)

Fig. 2 shows the parts selection and data types definition. We enter the URL of top page of BBC Country Profiles. WIKE lists the URLs of country pages. We select a typical country page. We define the names and data types for each selected part: country name part is CountryName of single text type, country map part is CountryMap of single object type, country facts part is Facts of continuous text type, leader’s photo part is LeaderPicture of single object type, and leader’s information part is LeaderInfo of single text type. Each selected part is represented by a node, and each node can be represented by its path from the root. We use the following form to save the path of a selected part:

\[
\text{body} : 0 : ID/N_1 : (O_1 - h \sim O_1 + h) : ID_1/N_2 : (O_2 - h \sim O_2 + h) : ID_2/.../N_{n-1} : (O_{n-1} - h \sim O_{n-1} + h) : ID_{n-1}/N_n : (O_n - h \sim O_n + h) : ID_n
\]

where, \( N_n \) is the name of the n-th node, \( O_n \) is the order of the n-th node among the sibling nodes, and \( N_{n-1} \) is the parent node of \( N_n \).
are still some country pages use the different layout. The defined paths and data types from only one country page can not satisfy the extraction from all the country pages.

WIKE tries extracting the partial information from all the country pages, and lists the country pages from which WIKE can not extract the correct partial information by using the defined paths and data types. The users can define the new paths and data types for one listed country page, and try the extraction again. The users continue this process until the partial information of all the country pages is extracted correctly.

We selected the parts and defined the data types from three typical country pages. All the defined paths and data types comprise the final extraction pattern of BBC Country Profiles.

2.4 Extraction Result

WIKE uses the extraction pattern to extract the partial information from the country pages. The extraction result is an XML-based document. WIKE returns the users a resulting table to show the extracted information.

A resulting table comprises two parts: field name and field value. Field name is the defined name of selected part and field value is the extracted value. In a resulting table, each column is a field and each row shows the extracted results of each country page, except that the first row displays the field names.

Fig. 3 is an extraction resulting table of BBC Country Profiles. The first column Link shows the URLs of country pages. The second column CountryName shows the country names. The third column CountryMap shows the URIs of country map pictures. The fourth column Facts shows the country facts. The fifth column LeaderPicture shows the URIs of leaders’ photos, and the sixth column LeaderInfo shows the leaders’ information.

3 Web Service Generation

A Web service is a system designed to support interoperable machine to machine interaction over a network, and executed on a remote system hosting the requested service. Usually, a real Web service can access to the server-side database and search for the data in tables to create the response results. A virtual Web service can be generated from the Web application by using the extraction function of WIKE. As shown in Fig. 4, in the proxy server, the extraction patterns are used to extract the partial information, and the resulting tables are generated. By a designed interface, the users send the requests to proxy server and get the responses from the resulting tables.

Figure 4. The Web Service Generated by Using Extraction Function of WIKE

In the proxy server, a standard RESTful Web service interfaces is used between the users and the generated Web service. Through the interface, the Web service gets the request from the user, searches for the desired information in the resulting table, and returns the corresponding values to user in XML format. If the structure of desired field is continuous occurrence, the corresponding response data is the value list. The generated Web service works like a real Web service, and can be integrated with other Web services to construct a Mashup system as shown in Fig. 5.

As well as the Web pages with static URLs, the dynamically generated Web pages can also be used to generate the Web services. Fig. 6 shows a response of news search from the generated CNN News [3] search Web service.

4 Related Work and Evaluation

A number of approaches have been proposed to analyze the Web page structures of Web applications with the purpose of Web information/knowledge extraction and Web service generation.
ANDES [7] and PSO [9] extract the information from Web pages by paths. Crunch [4] is an HTML tag filter to retrieve the contents from the Web pages. These systems need the users to find out the paths or create the tag filters from the HTML documents by hands. IEPAD [2] and Web Service Gateway [5] are difficult for the users to develop the personalized extraction. HTML2RSS [8] is limited to extract the special data structures such as time-series items from blog, BBS, chats and mailing lists. GridXSLT [6] needs the programming of users.

Compared to the existing work, WIKE realizes the extraction of almost all kinds of partial information such as text, object and link. It provides a visual editor for easy parts selection and data types definition. The extraction pattern creation or partial information extraction does not need programming. The similar paths are used in node extraction, and the users do not need to do parts selection and data types definition for each Web page.

However, this demo still depends on some manual work during the Web service interface configuration, and is not robust enough when the Web applications change the layout of Web pages. If the layout of Web page is changed totally and the nodes can not be found by the similar paths within the deviation range, the extraction would fail partially or completely, and the users have to create a new extraction pattern.

5 Conclusion and Future Work

In this demo, we have presented WIKE, a Web information/knowledge extraction system for Web service generation. WIKE provides a visual editor for easy extraction. The users can select the parts, define the data types, and create the extraction pattern without programming. The typed data is extracted and the extraction result is the XML-based document. The Web service is generated based on the extraction function of WIKE. Through the designed interface, the generated Web service can be integrated with other Web services to construct a Mashup system.

As future work, we will enhance the extraction function of WIKE in order to extract the exact parts from the updated Web pages. Moreover, we will modify WIKE to decrease the manual work during the Web service interface configuration.

References