

Article ID:

Web Database Query Interface Annotation Based on User Collaboration

□ LIU Wei, LIN Can, MENG Xiaofeng
School of Information, Renmin University of China, Beijing
100872, Beijing, China

Abstract: A vision based query interface annotation method is used to relate attributes and form elements in form-based web query interfaces, this method can reach accuracy of 82%. And a user participation method is used to tune the result; user can answer "yes" or "no" for existing annotations, or manually annotate form elements. Mass feedback is added to the annotation algorithm to produce more accurate result. By this approach, query interface annotation can reach a perfect accuracy.

Key words: web database; data integration; data extraction

CLC number: TP393.09

Received date: 2006-03-25

Foundation item: Supported by the Natural Science Foundation of China (60573091, 60273018)

Biography: LIU Wei (1976-), Male, Ph.D. candidate, and Research Direction: Web Data Integration. Email: gue2@ruc.edu.cn

0 Introduction

The Web has been rapidly deepened with the prevalence of online databases. These databases are Web accessible through form-based query interfaces. We call these online databases "Web Database". A study^[1] estimated 43 000-96 000 such search sites (and 550 billion content pages) on the Web. A recent survey^[2] in April 2004 estimated 450 000 Web databases. Current crawlers cannot effectively query databases, so such databases are invisible to search engines.

The form-based query interfaces are designed for user interaction, providing a programmatic way to access web databases under query interfaces and integrate search system over Web databases has become an application in high demand. Some Web databases have provided Web Services, but most Web databases not. To enable effective access to Web databases without Web Services, there are many researchers devoting themselves to this area, and a great deal of research works have been proposed to address the related issues^[3,4,5,6]. Among these issues, query interfaces annotation is one of the most important topics. It is the technique to relate attributes and elements.

Web 2.0^[7] is design patterns and business models for the next generation of software and web; one important principle of Web 2.0 user collaboration. Based on the idea of Web 2.0, user participation is used to tune the annotation accuracy in this paper.

1 Web Service Building System Architecture

Fig 1 is the high level architecture of the Web Service Building System. The goal of this system is to provide stimulant web service for programmatic usage of the web database. Two major parts are query interface parsing and annotation, result page record extraction and annotation. This paper put focus on the first subsystem.

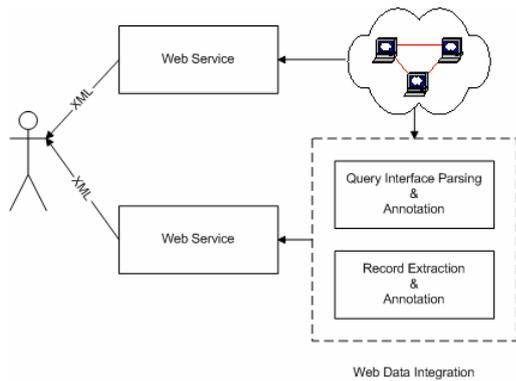


Fig.1 High level architecture of web service building system

Fig 2 is the block diagram of query interface parsing and annotation subsystem. The major process including:

- ① Fetch the URL has form-based query interfaces
- ② Parse out forms on the webpage.
- ③ Vision based query interface annotation
- ④ User anticipation to tune the annotation result
- ⑤ Put annotated result query interface repository for future reuse.

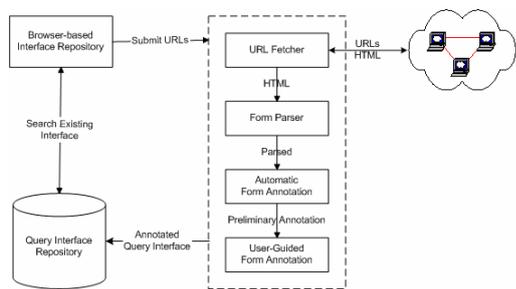


Fig.2 block diagram of query interface annotation subsystem

The second subsystem is based on SG-Wrap^[8,9], a Schema-Guided wrapper for HTML pages. After query interface annotation, user can use online SG-Wrap service to annotate the result page. After these 2 steps, Stimulant web services can be provided.

2 Query Interface Annotation

2.1 Web Query Interface Introduction

Web Query Interface is for the purpose to submit queries. Most modern web sites are powered by backend databases, queries submitted by user are constructed as SQL to fetch data from backend database and send it back to user mashed with html tags. Query interfaces are presented by form tags.

These form elements represent the query capability of the query interface. If there is a text element stand for author in a book search system, then the Author can be used as search condition. Besides elements, the annotation text for form elements is important.

2.2 Query Interface Parsing

HTML is a presentation language, query interfaces is expressed with conjunction of semantic information, formatted information, layout information. Query interface parsing is to extract the query capability from form-based query interfaces; it is to parse out form elements including `<form>`, `<input>`, `<textarea>`, `<select>`, `<option>`, `<button>`.

The result user got after he has sent the request is only based on the "real" values send via HTTP protocol. With the prevalence of Rich Internet Applications, and client side technologies such as Javascript, AJAX, etc, the query interface is becoming more and more complex and flexible. But the request send to the server based on HTTP protocol is only simple name-value pairs. The purpose of Interface Parsing is to reduce the complexity and diversity of different query interfaces.

Query interface parsing generate a XML format file for each form which describe the query capability and request approach of query interface. Fig 3 is a sample parsing result.

This file also contains the element level information, for each element, it records its name, type, default value, value list, and annotation. Except the annotation, other values can be got almost fully automatically. The method to find out annotation for each form element will be discussed in the next section.

```

- <form>
  <action dynamic="yes">http://portal.acm.org/results.cfm?
  coll=Portal&dl=Portal&CFID=72294949&CFTOKEN=11654175</action>
  <baseurl>http://portal.acm.org/</baseurl>
  <method>post</method>
- <elements>
  - <element>
    <name>whichDL</name>
    <default>acm</default>
    <annotation>acm: ACM Digital Library; guide: The Guide</annotation>
    <type>radio</type>
  </element>
  - <element>
    <name>parser</name>
    <default>internet</default>
    <annotation />
    <type>hidden</type>
  </element>
  - <element>
    <name>query</name>
    <default />
    <annotation>Keyword</annotation>
    <type>text</type>
  </element>
</elements>
</form>

```

Fig 3 a sample query interface parsing result

2.3 Vision Based Query Interface Annotation

Form-based query interface not only present the query capabilities, but also have semantic information. The most important semantic information is the annotation of each form element. For example, there is a select box element named "field-age" in Amazon's book search interface, but what is "field-age" stand for? When user sees the query interface, he will see that there is an annotated text "Reader Age" at the left side of this select box. But for programmatic usage, the relationship of the annotated text and corresponding form element lost. The purpose of query interface annotation is to group text and form elements together, to give semantic information to program.

Based on the idea of Liu Yin^[10], a vision based query interface annotation method is used to solve this problem. The basic idea is to obtain a visual representation of the form elements and continues text within a form. For example, a query interface shown in Fig 4.

The image shows a search filter section with the following elements:

- Used Only:** A checkbox that is currently unchecked.
- Format:** A dropdown menu with "All formats" selected.
- Reader age:** A dropdown menu with "All ages" selected.
- Language:** A dropdown menu with "All languages" selected.
- Publication date:** A dropdown menu with "All dates" selected, followed by a text input field containing "(e. g. 1999)".
- Sort results by:** A dropdown menu with "Relevance" selected.

Fig 4 sample query interface from amazon.com

The texts and elements within this interface are transformed into a block representation. Each text

snippet and element is represented as a rectangle, their geometrical information, including width, height, x-offset, and y-offset can be calculated based on the DOM model of modern browser. The interface in Fig 4 is represented as Fig 5.

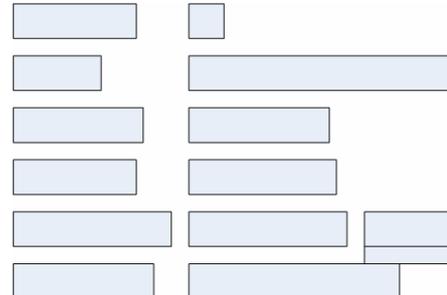


Fig 5 block presentation of query interfaces in Fig 4

Blocks in Fig 5 fall into 2 categories: text blocks and element blocks. The annotation is to find out the best correlation between text blocks and element blocks.

Suppose there are m text blocks $T_1, T_2 \dots T_m$ and n element blocks $E_1, E_2 \dots E_n$. The distance between T_i and E_j is defined as the shortest distance of the two rectangles:

$$D(T_i, E_j) = SD(T_i, E_j)$$

The total distance of one grouping is:

$$D(T_i, E_j) = \sum SD(T_i, E_j)$$

The grouping achieve the lowest D is the best grouping.

2.4 User-Guided Annotation Tuning

The above vision based annotation can achieve the precision of about 82%. In order to put this system into real world use, user anticipation is used to tuning the annotation result.

There are 2 types of user anticipation, on is answer whether a text and element grouping is correct or not, another is to manually annotate some form elements. Some form elements don't have annotations, but user can get it from context.

For both type of user anticipation, a 2 column web page is shown to users, the left side is the parsed and annotated query interface and the right side is the

original query interface. User click on each text or form element at one side, the corresponding part in another side will be highlighted.

In the answering method, user only needs to answer yes or no about a given grouping. Then the user feedback is added to the distance function, the best grouping is recalculated. Take user feedback into consideration, here $\sum n$ means number of answer “no”,

$\sum y$ means number of answer “yes”.

$$D(T_i, E_j) = \frac{SD(T_i, E_j) \times \sum n}{\sum y + \sum n}$$

Another user anticipation method is to let user input the annotation text manually. In the automatic annotation step, some form elements will not have annotation. User can input his annotation, after the first input, others can vote this as yes or no, and they can also change it.

The query interface annotation subsystem is based on user anticipation. The more users use this service, the better the service will be. An important issue in the approach is how to encourage user to use this service.

There are two assumptions in the user anticipation method.

- ① The service is useful and can increase the productivity of users.
- ② The web query interface annotation task is tedious and hard for users to complete by themselves.

In the web service building system, there is a large scale prepared web sites whose query interface and result page has been well parsed and annotated, Users can use them directly. In order to use this repository, user registering is required; each register user can use 10 pre-generated services, if he wants to use more, he must submit new websites or help the system the check the correctness of existing websites.

3 Conclusion

Interface parsing and annotation is the one major part of the web service building system, another major

part is result page record extraction and annotation. It is based on SGWrap^[8] to extract the attributes of result page records and annotate these items. The interface annotation work combines automatic annotation method and user anticipation to achieve much higher accuracy.

References

- [1] Bergman Michael K. The Deep Web: Surfacing Hidden Value[R]. *The Journal of Electronic Publishing* 7 (1)
- [2] Chang Kevin Chen-Chuan. He Bin, Li Chengkai, et al. Structured databases on the web: Observations and implications[R]. *SIGMOD Record*, 2004, 33(3):61—70
- [3] He Hai, Meng Weiyi, Yu Clement T, Wu Zonghuan. Wise-integrator: An automatic integrator of web search interfaces for e-commerce[C]. *Proceeding of the 28th Conference on Very Large Data Bases*, 2003:357-368.
- [4] He Bin, Chang Kevin Chen-Chuan. Statistical schema matching across web query interfaces[C]. *Proceeding of SIGMOD*, 2003.
- [5] Wu Wensheng, Yu Clement T, Doan AnHai, Meng Weiyi. An interactive clustering-based approach to integrating source query interfaces on the Deep Web[C]. *Proceeding of SIGMOD*, 2004: 95-106.
- [6] He Bin, Chang Kevin Chen-Chuan, Han Jiawei. Discovering complex matchings across web query interfaces: A correlation mining approach[C]. *Proceeding of Knowledge Discovery and Data Mining*, 2004.
- [7] Tim O'Reilly. What is Web 2.0: Design Patterns and Business Models for the Next Generation of Software [EB/OL]. [2005-09-30]. <http://www.oreillyn.com/pub/a/oreilly/tim/news/2005/09/30/what-is-web-20.html>
- [8] Meng Xiaofeng, Lu Hongjun, Wang Haiyan and Gu Mingzhe. Data extraction from the web based on pre-defined schema[J]. *Journal of Computer Science and Technology*, 2002, 17(4):377-388.
- [9] Yi Lei, Hu Dongdong, Yu Juntao, et al. OrientI: A Strategy of Web Information Integration. *Wuhan University Journal of Natural Sciences*, 2004.
- [10] Liu Yin, Liu Wenyin, Jiang Changjun. User Interest Detection on Web Pages for Building Personalized Information Agent. *Proceeding of International Conference on Web-Age Information Management*, 2004:280-287.