International Journal of Software & Hardware Research in Engineering
ISSN No: 2347-4890 Volume 2 Issue 12, December-

Annotating Search Record from Web Databases

Prasad B. Dhore[1], Rajesh B. singh[2]

1*Department of Computer Engineering, Nutan Maharashtra Vidya Polytechnic, Talegaon 410507, Pune, Maharashtra, India.*
2*Associate Professor, Department of Computer Engineering, Sinhgad Institute of Technology, Lonavala 410401, Pune, Maharashtra, India.*
dhoreprasad@yahoo.com
rbs.sit@sinhgad.edu

Abstract **-** *The paper investigates techniques for extracting data from HTML sites through the use of automatically generated wrappers. To automate the wrapper generation and the data extraction process, the paper develops In this paper, we present an automatic annotation approach that first aligns the data units on a result page into different groups such that the data in the same group have the same semantic. So Our experiments indicate that the proposed approach is highly effective.*

Keywords*: HTML*

I: INTRODUCTION

Data extraction from HTML is usually performed by software modules called wrappers. Early approaches to wrapping Web sites were based on manual techniques [2, 9, 17, 4, 11]. A key problem with manually coded wrappers is that writing them is usually a difficult and labor intensive task, and that by their nature wrappers tend to be brittle and difficult to maintain. A large portion of the deep web is database based, i.e., for many search engines, data encoded in the returned result pages come from the underlying structured databases. Such type of search engines is often referred as Web databases (WDB).A data unit is a piece of text that semantically represents one concept of an entity. It corresponds to the value of a record under an attribute. It is different from a text node which refers to a sequence of text surrounded by a pair of HTML tags. There is a high demand for collecting data of interest from multiple WDBs. While most existing approaches simply assign labels to each HTML text node, we thoroughly analyze the relationships between text nodes and data units.

We perform data unit level annotation. A clustering-based shifting technique to align data units into different groups so that the data units inside the same group have the same semantic. Instead of using only the DOM tree or other HTML tag tree structures of the SRRs to align the data units (like most current methods do), our approach also considers other important features shared among data units, such as their data types (DT), data contents (DC), presentation styles (PS),and adjacency (AD) information.

II: RELATED ARTICLES

R.Saranya and Dr.J.Komalalakshmi (1) develop proposes a domain independent modified clustering technique with String similarity measure for rectifying the data annotation and alignment problem to great extent. The process of precision and recall for search results shows the usability and reusability of the books (domain) found in the respective websites. The Experimental Results prove that, the individual website comparison and calculation among all selected websites eliminates the data alignment problem.

V.kalyan Deepak, N.V.Rajeesh Kumar (2) develop an automatic annotation approach that first aligns the data units on a result page into different groups such that the data in the same group have the same semantic. Then, for each group we annotate it from different aspects and aggregate the different annotations to predict a final annotation label for it. An annotation wrapper for the search site is automatically constructed and can be used to annotate new result pages from the same web database.

P.V.Praveen Sundar (3) develop to classifies a region in the web page according to similar data object which emerge frequently in it. This involves transformation of unstructured data into structured data that can be stored and analyzed in a central local database. The existing system develops a data extraction and alignment method known as combining tag and value similarity (CTVS), which identifies the query result records (QRRs) by extracting the data from query result page and segment them. Those segmented QRRs are aligned into a table where same attribute data values are put into the same column.

IJournals

© 2014, IJournals All Rights Reserved

III: SUMMARY OF EXISTING SYSTEM

In this existing system, If ISBNs are not available, their titles and authors could be compared. The system also needs to list the prices offered by each site. The system needs to know the semantic of each data unit. Unfortunately, the semantic labels of data units are often not provided in result pages. For instance, no semantic labels for the values of title, author, publisher, etc., are Having semantic labels for data units is not only important for the above record linkage task, butalso for storing collected SRRs into a database table.
In this system data unit is a piece of text that semantically represents one concept of an entity. It corresponds to the value of a record under an attribute. It is different from a text node which refers to a sequence of text surrounded by a pair of HTML tags. It describes the relationships between text nodes and data units in detail. In this paper, we perform data unit level annotation. There is a high demand for collecting data of interest from multiple WDBs.

IV: PROPOSED SYSTEM

We utilize the integrated interface schema (IIS) over multiple WDBs in the same domain to enhance data unit annotation. The six basic annotators; each annotator can independently assign labels to data units based on certain features of the data units. We also employ a probabilistic model to combine the results from different annotators into a single label. Construction of annotation wrapper for any given WDB. The wrapper can be applied to efficiently annotating the SRRs retrieved from the same WDB with new queries.

In proposed system, we consider how to automatically assign labels to the data units within the SRRs returned from WDBs. Given a set of SRRs that have been extracted from a result page returned from a WDB. The data retrieval from the web site and its storage as collected data is the result of the web mining process in the data mining. The extracted data will be used for the further research in all areas. extracting
data from HTML sites through the use of automatically generated wrappers. To automate the wrapper generation and the data extraction process, the paper develops a novel technique to compare HTML pages and generate a wrapper based on their similarities and differences.

V: PROPOSED SYSTEM ARCHITECTURE

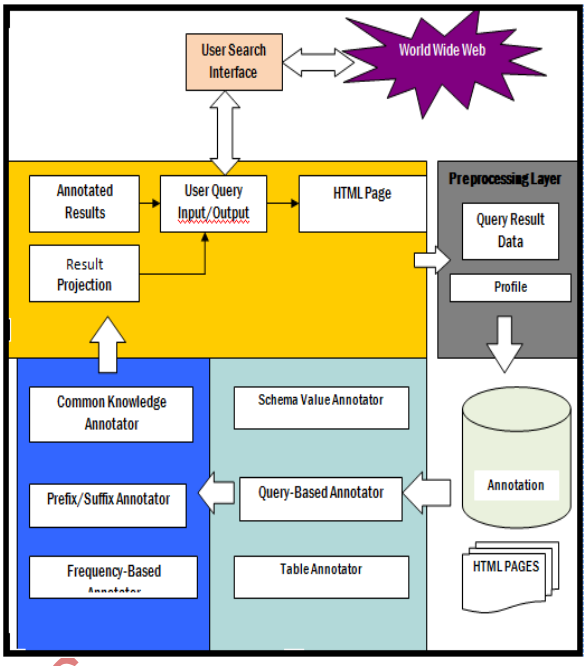

Figure: System Architecture

VI: IMPLEMENTATION

There are five modules:

- Basic Annotators
- Query-Based Annotator
- Schema Value Annotator
- Common Knowledge Annotator
- Combining Annotators

A: Basic Annotators

In a returned result page containing multiple SRRs, the data units corresponding to the same concept (attribute) often share special common features. And such common features are usually associated with the data units on the result page in certain patterns. Based on this observation, we define six basic annotators to label data units, with each of them considering a special type of patterns/features. Four of these annotators (i.e., table annotator, query-based annotator, intext

IJournals

© 2014, IJournals All Rights Reserved

prefix/suffix annotator, and common knowledge annotator) are similar to the annotation heuristics

B: Query-Based Annotator

The basic idea of this annotator is that the returned SRRs from aWDBare always related to the specified query. Specifically, the query terms entered in the search attributes on the local search interface of the WDB will most likely appear in some retrieved SRRs. For example, query term "machine" is submitted through the Title field on the search interface of the WDB and all three titles of the returned SRRs contain this query term. Thus, we can use the name of search field Title to annotate the title values of these SRRs. In general, query terms against an attribute may be entered to a textbox or chosen from a selection list on the local search interface. Our Query-based Annotator works as follows: Given a query with a set of query terms submitted against an attribute A on the local search interface, first find the group that has the largest total occurrences of these query terms and then assign gn(A) as the label to the group.

C: Schema Value Annotator

Many attributes on a search interface have predefined values on the interface. For example, the attribute Publishers may have a set of predefined values (i.e., publishers) in its selection list. More attributes in the IIS tend to have predefined values and these attributes are likely to have more such values than those in LISs, because when attributes from multiple interfaces are integrated, their values are also combined. Our schema value annotator utilizes the combined value set to perform annotation.

The schema value annotator first identifies the attribute Aj that has the highest matching score among all attributes and then uses gn(Aj) to annotate the group Gi. Note that multiplying the above sum by the number of nonzero similarities is to give preference to attributes that have more matches (i.e., having nonzero similarities) over those that have fewer matches. This is found to be very effective in improving the retrieval effectiveness of combination systems in information retrieval

D: Common Knowledge Annotator

Some data units on the result page are self-explanatory because of the common knowledge shared by human beings. For example, "in stock" and "out of stock" occur in many SRRs from e-commerce sites. Human users understand that it is about the availability of the product because this is common knowledge. So our common knowledge annotator tries to exploit this situation by using some predefined common concepts.

Each common concept contains a label and a set of patterns or values. For example, a country concept has a label "country" and a set of values such as "U.S.A.," "Canada," and so on. It should be pointed out that our common concepts are different from the ontologies that are widely used in some works in Semantic Web. First, our common concepts are domain independent. Second, they can be obtained from existing information resources with little additional human effort.

E: Combining Annotators

Our analysis indicates that no single annotator is capable of fully labeling all the data units on different result pages. The applicability of an annotator is the percentage of the attributes to which the annotator can be applied. For example, if out of 10 attributes, four appear in tables, then the applicability of the table annotator is 40 percent. The average applicability of each basic annotator across all testing domains in our data set. This indicates that the results of different basic annotators should be combined in order to annotate a higher percentage of data units. Moreover, different annotators may produce different labels for a given group of data units. Therefore, we need a method to select the most suitable one for the group. Our annotators are fairly independent from each other since each exploits an independent feature.

VII: CONCLUSION

In this paper, we studied the data annotation problem and proposed a multiannotator approach to automatically constructing an annotation wrapper for annotating the search result records retrieved from any given web database. A special feature of our method is that, when annotating the results retrieved from a web database, it utilizes both the LIS of the web database and the IIS of multiple web databases in the same domain. We also explained how the use of the IIS can help alleviate the local interface schema inadequacy problem and the inconsistent label problem. In this paper, we also studied the automatic data alignment problem. Accurate alignment is critical to achieving holistic and accurate annotation. Our method is a clustering based shifting method utilizing richer yet automatically obtainable features. This method is capable of handling a variety of relationships between HTML text nodes and data units, including one-to-one, one-to-many, many-to-one relationship.

VIII: REFERENCES

[1] A. Arasu and H. Garcia-Molina, "Extracting Structured Data from Web Pages," Proc. SIGMOD Int'l Conf. Management of Data, 2003.

[2] L. Arlotta, V. Crescenzi, G. Mecca, and P. Merialdo, "Automatic Annotation of Data Extracted from Large

IJournals

© 2014, IJournals All Rights Reserved

Web Sites," Proc. Sixth Int'l Workshop the Web and Databases (WebDB), 2003.

[3] P.V.Praveen Sundar, "Towards Automatic Data Extraction Using Tag and Value Similarity Based on Structural -Semantic Entropy",International Journal of Advanced Research in Computer Science and Software Engineering" Volume 3 Issue 4 Edition, pp: 226-231. April 2013

[4] P. Chan and S. Stolfo, "Experiments on Multistrategy Learning by Meta- Learning," Proc. Second Int'l Conf. Information and Knowledge Management (CIKM), 1993.

[5] W. Bruce Croft, "Combining Approaches for Information Retrieval," Advances in Information Retrieval: Recent Research from the Center for Intelligent Information Retrieval, Kluwer Academic, 2000.

[6] V. Crescenzi, G. Mecca, and P. Merialdo, "RoadRUNNER: Towards Automatic Data Extraction from Large Web Sites," Proc. Very Large Data Bases (VLDB) Conf., 2001.

[7] S. Dill et al., "SemTag and Seeker: Bootstrapping the Semantic Web via Automated Semantic Annotation," Proc. 12th Int'l Conf. World Wide Web (WWW) Conf., 2003.

[8] H. Elmeleegy, J. Madhavan, and A. Halevy, "Harvesting Relational Tables from Lists on the Web," Proc. Very Large Databases (VLDB) Conf., 2009.

[9] D. Embley, D. Campbell, Y. Jiang, S. Liddle, D. Lonsdale, Y. Ng, and R. Smith, "Conceptual-Model-Based Data Extraction from Multiple- Record Web Pages," Data and Knowledge Eng., vol. 31, no. 3, pp. 227- 251, 1999.

[10] D. Freitag, "Multistrategy Learning for Information Extraction," Proc. 15th Int'l Conf. Machine Learning (ICML), 1998.

[11] D. Goldberg, Genetic Algorithms in Search, Optimization and Machine Learning. Addison Wesley, 1989.

IJournals

© 2014, IJournals All Rights Reserved