International Journal of Science and Research (IJSR)
ISSN (Online): 2319-7064
Index Copernicus Value (2013): 6.14 | Impact Factor (2014): 5.611

Survey of Web Database Clustering Techniques

Chaitali R. Hokam[1], Vaishali P. Suryawanshi[2]

[1]ME student, MITCOE, Pune-38, India

[2]Assistant Professor, MITCOE, Pune-38, India

Abstract: ***Nowadays databases have become web accessible through HTML form-based search interfaces The data units returned are encoded into the result pages dynamically for human browsing. For the encoded data units to be machine processable , which is essential for many applications such as deep web data collection and Internet comparison shopping, they need to be extracted out and assigned meaningful labels. Deep Web contents are accessed by queries submitted to Web databases and the returned data records are enwrapped in dynamically generated Web pages. Extracting structured data from deep Web pages is a challenging problem due to the underlying intricate structures of such pages. Author proposes some automated annotating approaches to improving search method on basis similarities in data units. They can form groups of data units and label them with semantic names and can store result for future analysis also.***

Keywords: Data alignment, data annotation, web database, wrapper generation

1. Introduction

The Web has become the preferred medium for many database applications, such as e-commerce and digital libraries. These applications store information in huge databases that user‟s access, query, and update through the Web. Database-driven Web sites have their own interfaces and access forms for creating HTML pages on the fly. Web database technologies define the way that these forms can connect to and retrieve data from database servers. The number of database-driven Websites are increasing exponentially, and each site is creating pages dynamically pages that are hard for traditional search engines to reach. Such search engines crawl and index static HTML pages; they do not send queries to Web databases.

The encoded data units to be machine processable, which is essential for many applications such as deep web data collection and Internet comparison shopping, they need to be extracted out and assigned meaningful labels.

The explosive growth and popularity of the World Wide Web has resulted in a huge amount of information sources on the Internet. However, due to the heterogeneity and the lack of structure of Web information sources, access to this huge collection of information has been limited to browsing and searching. Sophisticated Web mining applications, such as comparison shopping robots, require expensive maintenance to deal with different data formats. To automate the translation of input pages into structured data, a lot of efforts have been devoted in the area of information extraction (IE). Unlike information retrieval (IR), which concerns how to identify relevant documents from a document collection, IE produces structured data ready for post processing, which is crucial to many applications of Web mining and searching tools.

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 typical result page returned from a WDB has multiple search result records(SRRs). Each SRR contains multiple data unit search of which describes one aspect of a real-world entity. In this paper, 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. In this paper, author perform data unit level annotation There is a high demand for collecting data of interest from multiple WDBs. For example, once a book comparison shopping system collects multiple result records from different book sites, it needs to determine whether any two SRRs refer to the same book.

The authors propose 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.

2. Literature Survey

1] Annotating Search Results from Web Databases [1]

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 typical result page returned from a WDB has multiple search result records (SRRs). Each SRR contains multiple data units each of which describes one aspect of a real-world entity. In this paper, 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. In this paper, author perform data unit level annotation.

2] Vision based data extraction method [2]

In this paper, the author explores the visual regularity of the data records and data items on deep Web pages and propose

a novel vision-based approach, Vision-based Data Extractor (ViDE), to extract structured results from deep Web pages automatically. ViDE is primarily based on the visual features human users can capture on the deep Web pages while also utilizing some simple non visual information such as data types and frequent symbols to make the solution more robust. ViDE consists of two main components, Vision based Data Record extractor (ViDRE) and Vision-based Data Item extractor (ViDIE). By using visual features for data extraction, ViDE avoids the limitations of those solutions that need to analyze complex Web page source files.

Figure 1: Vision based data extraction method

3] Harvesting relational tables from lists on the web [3]

A large number of web pages contain data structured in the form of "lists". Many such lists can be further split into multi-column tables, which can then be used in more semantically meaningful tasks. However, harvesting relational tables from such lists can be a challenging task. The lists are manually generated and hence need not have well-defined templates they have inconsistent delimiters (if any) and often have missing information. Author propose a novel technique for extracting tables from lists. The technique is domain independent and operates in a fully unsupervised manner. They first use multiple sources of information to split individual lines into multiple fields and then, compare the splits across multiple lines to identify and fix incorrect splits and bad alignments. In particular, they exploit a corpus of HTML tables, also extracted from the web, to identify likely fields and good alignments. For each extracted table, author compute an extraction score. Author conducted an extensive experimental study using both real web lists and lists derived from tables on the web. The experiments demonstrate the ability of our technique to extract tables with high accuracy. In addition, author applied the technique on a large sample of about 100,000 lists crawled from the web.

4] Automatic integration of Web search interfaces with WISE-Integrator [4]

An increasing number of databases are becoming Web accessible through form-based search interfaces, and many of these sources are database-driven e-commerce sites. It is a daunting task for users to access numerous Web sites individually to get the desired information. Hence, providing a unified access to multiple e-commerce search engines selling similar products is of great importance in allowing users to search and compare products from multiple sites with ease. One key task for providing such a capability is to integrate the Web search interfaces of these e-commerce search engines so that user queries can be submitted against the integrated interface. Currently, integrating such search interfaces is carried out either manually or semi automatically, which is inefficient and difficult to maintain. In this paper, the author presented a WISE-Integrator, a tool that performs automatic integration of Web Interfaces of Search Engines. WISE-Integrator explores a rich set of special meta information that exists in Web search interfaces and uses the information to identify matching attributes from different search interfaces for integration. It also resolves domain differences of matching attributes. In this paper, author has also discuss how to automatically extract information from search interfaces that is needed by WISE-Integrator to perform automatic interface integration.

5] Fully Automatic Wrapper Generation For Search Engines [5]

When a query is submitted to a search engine, the search engine returns a dynamically generated result page containing the result records, each of which usually consists of a link to and/or snippet of a retrieved Web page. In addition, such a result page often also contains information irrelevant to the query, such as information related to the hosting site of the search engine and advertisements. In this paper, author has presented a technique for automatically producing wrappers that can be used to extract search result records from dynamically generated result pages returned by search engines. Automatic search result record extraction is very important for many applications that need to interact with search engines such as automatic construction and maintenance of meta search engines and deep Web crawling. The novel aspect of the proposed technique is that it utilizes both the visual content features on the result page as displayed on a browser and the HTML tag structures of the HTML source file of the result page.

3. Automated Annotation Process

The automatic annotation solution consist of 3 phases:

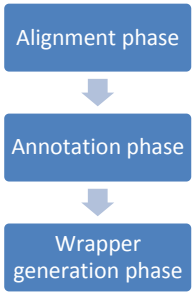

fig2: Automated Annotation Process

The following modules describes the automated annotation process.

1) Data Alignment
2) Data Annotation
3) Automatic Wrapper Generation

1. Data Alignment:

- In this module, first identify all data units in the search records and then organize them into different groups with each group corresponding to a different concept the result of this module with each column containing data units of the same concept across all search records.
- Grouping data units of the same meaning can help identify the common patterns and features among these data units.
- These common features are the basis of our annotators.

2. Data Annotation:

- Author introduces multiple basic annotators with each exploiting one type of features.
- Every basic annotator is used to produce a label for the units within their group holistically, and a probability model is adopted to determine the most appropriate label for each group.

3. Automatic Wrapper Generation:

- In this module author generated an annotation rule that describes how to extract the data units of this concept in the result page and what the appropriate meaning annotation should be.
- The rules for all aligned groups, collectively, form the annotation wrapper for the corresponding WDB, which can be used to directly assign label the data retrieved from the same WDB in response to new queries without the need to perform the above tow modules again.
- As such, annotation wrappers can perform annotation quickly, which is essential for online applications.

4. Alignment Algorithm

In this paper, the information arrangement calculation is focused around the supposition that traits show up in the same request over all SRRs on the same result page, in spite of the fact that the SRRs may contain diverse sets of credits (because of missing qualities). This is genuine when all is said in done on the grounds that the SRRs from the same WDB are typically produced by the same layout program. Accordingly, author can thoughtfully consider the SRRs on a result page in a table configuration where each one column speaks to one SRR and each one cell holds an information unit (or unfilled if the information unit is not accessible). Each one table section, in this work, is alluded to as an arrangement gathering, containing at most one information unit from every SRR. If an alignment group contains all the data units of one concept and no data unit from other concepts, call this group well-aligned. The goal of alignment is to move the data units in the table so that every alignment group is well aligned, while the order of the data units within every SRR is preserved. Data alignment algorithm is given as follows.

Alignment algorithm has following four steps:
Step 1: Merge text nodes: This step detects and removes decorative tags from each SRR to allow the text nodes corresponding to the same attribute merge into a single one.
Step 2: Align text nodes: After the merging aligns text nodes into different groups. So that same group has the same concepts.
Step 3: Split text nodes: In this step split the composite text nodes into separate data unit.
Step 4: Align data units: This is the last step for alignment in which separates each composite group into multiple aligned groups with each containing the data units of the same concept. The automatic annotation approach considers several types of data unit and text node features and makes annotation scalable and automatic. Basically three phases used for automatic annotation in which aligns the data units into different groups, labels each group and construct an annotation wrapper.

5. Applications

- Web data collection.
- Internet comparison shopping.

6. Conclusion

According to the survey, it automatically constructing an annotation wrapper for annotating the search result records retrieved from any given web database Basically three phases used for automatic annotation in which aligns the data units into different groups, labels each group and construct an annotation wrapper.

References

[1] W. Liu, X. Meng, and W. Meng, “ViDE: A Vision-Based Approach for Deep Web Data Extraction,” IEEE Trans. Knowledge and Data Eng., vol. 22, no. 3, pp. 447-460, Mar. 2010.
[2] Yiyao Lu, Hai He, Hongkun Zhao, WeiyiMeng, and Clement Yu “Annotating Search Results from Web Databases,” IEEE Trans. Knowledge and Data Eng ., vol. 25, no. 3, pp514-527, Mar 2013.
[3] HazemElmeleegy Purdue University hazem@cs.purdue.edu, JayantMadhavan Google Inc. jayant@google.com, Alon Halevy Google Inc. halevy@google.com,"Harvesting relational tables from lists on the web", VLDB ‟09, August 2428, 2009, Lyon, France
[4] H. He, W. Meng, C. Yu, and Z. Wu, “Automatic Integration of Web Search Interfaces with WISE-Integrator,” VLDB J., vol. 13, no. 3, pp. 256-273, Sept. 2004.
[5] H. Zhao, W. Meng, Z. Wu, V. Raghavan, and C. Yu, “Fully Automatic Wrapper Generation for Search Engines,” Proc. Int‟l Conf. World Wide Web (WWW), 2005.
[6] H. Zhao, W. Meng, and C. Yu, “Mining Templates form Search Result Records of Search Engines,” Proc. ACM SIGKDD Int‟l Conf. Knowledge Discovery and Data Mining, 2007.
[7] D. Goldberg, Genetic Algorithms in Search, Optimization and Machine Learning. Addison Wesley, 1989.
[8] D. Cai, S. Yu, J. Wen, and W. Ma, “Extracting Content Structurefor Web Pages Based on Visual Representation,” Proc. Asia PacificWeb Conf. (APWeb), pp. 406-417, 2003.

[9] B. Liu, R.L. Grossman, and Y. Zhai, "Mining Data Records in Web Pages," Proc. Int'l Conf. Knowledge Discovery and Data Mining (KDD), pp. 601-606, 2003.
[10] J. Wang and F.H. Lochovsky, "Data Extraction and Label Assignment for Web Databases," Proc. 12th Int'l Conf. World Wide Web (WWW), 2003.