Available Online at www.ijcsmc.com

International Journal of Computer Science and Mobile Computing

A Monthly Journal of Computer Science and Information Technology

ISSN 2320–088X

IJCSMC, Vol. 4, Issue. 1, January 2015, pg.254 – 261

RESEARCH ARTICLE

Automatic Annotation Search from Web-Database

Mr. Kiran C.Kulkarni, Prof. S.M.Rokade

Department of Computer Engineering,S.V.I.T College Nashik ,Maharashtra & Savitribai Phule university of Pune, India
Department of Computer Engineering,S.V.I.T College Nashik ,Maharashtra & Savitribai Phule university of Pune, India
kiranc.kulkarni@gmail.com

Abstract— In this system, we address the problem of automatically extracting data objects from a given web site and assigning meaningful labels to the data. In this system we majorly look on the web sites that provide a complex HTML search form, other than keyword searching, for users to query the back-end databases. Solving this problem will allow the data both to be extracted from such web sites and its schema to be captured, which makes it easier to do further manipulation and integration of the data. This problem is for three reasons. First, the system deal with HTML searches forms, which are designed for human use. it makes difficult for html code to identify all the form elements and submit correct queries. Second, the wrapper generate for each html page needs to be more efficient enough to extract not only plain and nested structure data. Third, the generated wrap- per is usually based on HTML structure of the tags, which may never affect the real database structure, and the original database field names are generally not encoded in the web pages. In addition, for large scale data, the solution to this problem needs to be automatic and fast . The online shopping today having great popularity and rapid growth .The Web or internet has become the most important medium for many applications, such as e-commerce and digital libraries. Database-driven Web sites have their own interfaces and access method for creating HTML pages on the fly. Web database techniques define the various ways that can connect to and retrieve or access data from database servers. In this paper, we present an automatic annotation (assign label) approach that first aligns the data units on a result page into various groups such that the data in the same group have the same meaning. And then we assign labels to each of this group .An annotation cover for the search site is automatically constructed and can be used to assign label to new result pages from the semantic web.

Keywords— Data alignment, Data annotation, Wrapper-generation, Alignment

I. INTRODUCTION

Shopping system require more maintenance to deal with different data formats. To translate the input pages into structured data automatically lot of effort committed 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 The Web has become the preferred medium for many database applications, such. Database-driven sites have their own interfaces and access method for creating HTML pages on the fly. Web database techniques define the various ways that these forms can connect to and retrieve data from database servers. The number of database-driven Websites is increasing exponentially, and each site is creating pages dynamically pages that are hard for traditional search engines to reach. Internet comparison shopping, they need to be

extracted out and assigned meaningful annotation. The internet database 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 access to this huge collection of information has been limited to browsing and searching. Sophisticated Web mining applications, such as data ready for post processing, which is crucial to many applications of Web mining and searching tools. A results record 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 meaningfully represents one concept of an entity. It consists of 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, we 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. We 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.

II. LITERATURE SURVEY

STUDY OF EXISTING SYSTEMS/ TECHNOLOGIES

Web information extraction and annotation ie assigning labels have been an active research area in recent years. Many systems[18][20] rely on human users to mark the desired information on sample pages and label the marked data at the same time, and then the system can induce a series of rules(wrapper) to extract the same set of information on Web Pages from the same source. These systems are name as a wrapper induction system. Because of the supervised training and learning process, these system scan usually achieve high extraction accuracy. However, they suffer from poor scalability and are not suitable for applications[24] that need to extract information froma large number of web sources.

ANALYSIS OF EXISTING SYSTEMS/ TECHNOLOGIES

Embley et al.[8] utilize ontologies together with sev-eral heuristics to automatically extract data in multirecord documents and label them. However, ontologies for different area must be constructed manually.

Mukherjee et al.[25] exploit the presentation styles and the spatial locality of semantically related items, but its learning process for annotation is domain dependent.Moreover, a seed of instances of semantic concepts in a set of HTML documents needs to be hand labelled. These methods are not fully automatic.

The efforts to automatically construct wrappers are but the wrappers are used for data extraction only(not for annotation). We are aware of several works[2],[28],[30] which aim at automatically assigning meaningful labels to the data units in SRRs. Arlotta et al. basically annotate dataunits with the nearest labels on result pages.This method has limited applicability because many WDBsdo not encode data units with their labels on result pages. In ODE system[28] ontologies are first constructed using query interfaces and result pages from WDBs in the same Fig. Illustration of our three-phase annotation. The domain ontology is then used to assign labels toeach data unit on result page. After labelling, the data values with the same label are naturally aligned. This method is sensitive to the quality and completeness of the ontologies generated. DeLa[30] first uses HTML tags to align data units by filling them into a table through a regular expression based data tree algo-rithm. Then, it employs four heuristics to select a label for each aligned table column. The approach in performs attributes extraction and labelling simultaneously. How-ever, the label set is predefined and contains only a small number of values.

COMPARISON OF EXISTING SYSTEMS WITH PROPOSED SYSTEM

We align data units and annotate the ones within the same semantic group holistically. Data alignment is an important step in achieving accurate annotation and it is also used in[25] and [30] Most existing automatic data alignment tech-niques are based on one or very few features. The most frequently used feature is HTML tag paths (TP)The as-sumption is that the sub trees corresponding to two data units in different SRRs but with the same concept usually have the same tag structure. However, this assumption is not always correct as the tag tree is very sensitive to even minor differences, which may be caused by the need to emphasize certain data units or erroneous coding. Vi-DIE [21] usesvisual features on result pages to perform align-ment and it also generates an

alignment wrapper. But its alignment is only at text node level, not data unit level. The method in first splits each SRR into text segments. The most common number of segments is determined to be the number of aligned columns (attributes). The SRR with more segment sare then re split using the common number. For each Sopwith fewer segments than the com-mon number, each segment is assigned to the most similar aligned column. Our data alignment approach differs from the previous works in the following aspects.

First, our approach handles all types of relationships between text nodes and data units while existing approaches consider only some of the types (i.e., one-to-one or one-to-many).Second, we use a variety of features together, includingthe ones used in existing approaches, while existing approaches use significantly fewer features (e.g., HTML tag in), vi-sual features in). All the features that we use can be au-tomatically obtained from the result page and do not need any domain specific ontology or knowledge. Third, we in-troduce a new clustering-based shifting algorithm to per-form alignment.Among all existing researches, DeLa is the most similar to our work. But our approach is significantly different from DeLas approach.

First, is purely based on HTML tags, while ours uses other important features such as data type, text content, and adjacency information. Sec-ond, our method handles all types of relationships between text nodes and data units,whereas deals with only two of them (i.e., one-to-one and one-to-many). Third, DeLa and our approach utilize different search interfaces of WDBs for annotation. Ours uses an IIS of multiple WDBs in the same domain, whereas uses only the local interface schema (LIS) of each individual WDB. Our analysis shows that utilizing IISs has several benefits, including significantly alleviating the local interface schema inadequacy problem and the inconsistent label problem. Fourth, we signifi-cantly enhanced DeLas annotation method. Specifically, Among the six basic annotators in our method, two(i.e., schema value annotator (SA) and frequency-based anno-tator (FA)) are new (i.e., not used is DeLa), three (table annotator (TA), query-based annotator (QA) and common knowledge annotator (CA)) have better implementations than the corresponding annotation heuristics in DeLa, and one (in-text prefix/suffix annotator (IA)) is the same as a heuristic in DeLa. For each of the three annotators that have different implementations, the specific difference and the motivation for using a different implementation . Fur-thermore, it is not clear how DeLa combines its annotation heuristics to produce a single label for an attribute, while we employ a probabilistic model to combine the results of different annotators. Finally, DeLa builds wrapper for each WDB just for data unit extraction.In our approach, we construct an annotation wrapper describing the rules not only for extraction but also for assigning labels. To enable fully automatic annotation, the result page shave to be automatically obtained and the SRRs need to be automat-ically extracted.

In a meta search context, result pages are retrieved by queries submitted by users (some reformat-ting may be needed when the queries are dispatched to individual WDBs). In the deep web crawling context, result pages are retrieved by queries automatically generated by the Deep Web Crawler. We employ ViNTs Searching for semantically and visually similar images on the Web.And mining annotations from them.to extract SRRs from result pages in this work. Each SRR is stored in a tree structure with a single root and each node in the tree cor-responds to an HTML tag or a piece of text in the original page. With this structure, it becomes easy to locate each node in the original HTML page. The physical position information of each node on the rendered page, including its coordinates and area size, can also be obtained using ViNTs.

This system is an extension of our previous work. The following summarizes the main improvements of this paper over. First, a significantly more comprehensive discussion about the relationships between text nodes and data units is provided. Specifically, this paper identifies four rela-tionship types and provides analysis of each type, while only two of the four types (i.e., one-to-one and one-to-many)were very briefly mentioned in. Second, the align-ment algorithm is significantly improved. A new step is added to handle the many-to-one relationship between text nodes and data units. In addition, a clustering-shift algorithm is introduced in this paper to explicitly handle the one-to nothing relationship between text nodes and data units while the previous version has a pure clustering algorithm. With these two improvements, the new alignment algorithm takes all four types of relationships into consideration. Third, the experiment section is significantly different from the previous version.Moreover, the experiments on alignment and annotation have been redone based on the new data set and the improved alignment algorithm.

III.IMPLEMENTATION

Our automatic annotation solution consists of three phases as

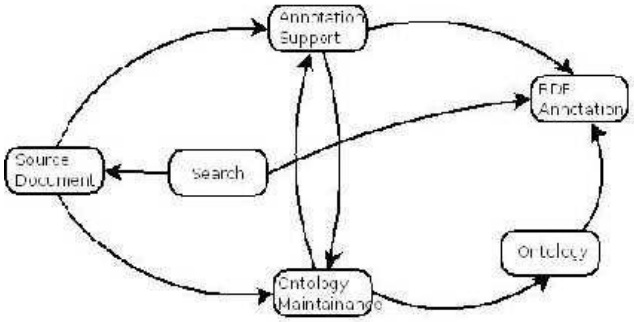


Fig. 1. Phases of automatic annotation solution

1.Extracts (automatically) text from a web-page into a table

2.Assigns labels in a table.

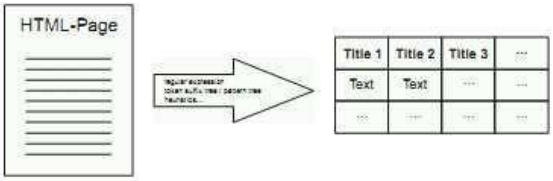


Fig. 2. Web data to table form

Phase 1 is the alignment phase, In this phase, we 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 phase 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 features are the basic of our annotator.

Phase 2 is the annotation phase we present multiple basic annotators with each utilize one type of features. Every basic annotator is used to produce a label for the units within their group a possibly model is adopted to determine the most appropriate label for each group.

Phase 3 is the annotation wrapper generation ,in this phase we generate 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 an- notation 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 two phases again. As such, annotation wrappers can perform label assign quickly, which is necessary for online applications.

We use multiannotator approach[31] to extract labels from various site.the six type annotator as follows

1.Table Annotator

In the table, each row represents an SRR. The table header, which indicates the meaning of each column, is usually located at the top of the table

2.Query Based Annotator

The basic idea of this annotator is that the returned SRRs from a WDB are 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

3. 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

4. Frequency BasedAnnotator

. In other words, the adjacent units have different occurrence frequencies. As argued in [1], the data units with the higher frequency are likely to be attribute names, as part of the template program for generating records, while the data units with the lower frequency most probably come from databases as embedded values.

5. In-Text Infix/Prefix Annotator

In some cases, a piece of data is encoded with its label to form a single unit without any obvious separator between the label and the value, but it contains both the label and the value

6.Common Knowledge Annotator

Some data units on the result page are self-explanatory because of the common knowledge shared by human beings.

IV. ALIGNMENT ALGORITHM

Data alignment algorithm is based on the assumption that attributes appear in two the same order across all SRRs on the same result page, although the SRRs may contain different sets of attributes (due to missing values). This is true because the SRRs from the same WDB are generated by the same template program. Thus, we can conceptually consider the SRRs on a result page in a table format where each row presents one SRR and each cell contain a data unit (or empty if the data unit is not available). Each table attribute (column), in our page, is referred to as an alignment group, containing at most one data unit from each SRR. If integrate group contains all the dataunits of one concept and no dataunit fromother concepts,we call this group well-aligned. The goal of alignment is to move the data units in the table so that every join group is well aligned, while the order of the dataunits within every SRR is preserved. Our dataalignment process consists of the following four steps as

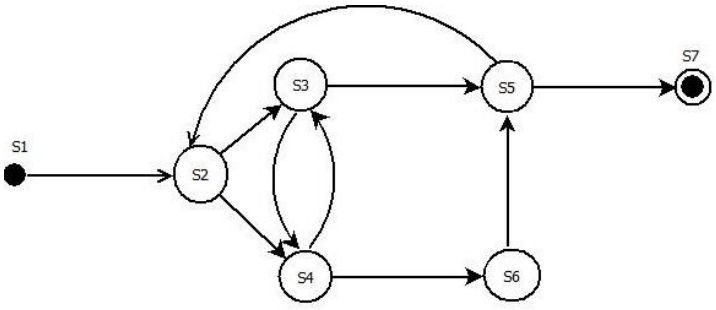
1.This step detects and removes decorative tags from each SRR to allow the textnodes corresponding to the same characteristic or attribute (separated by decorative tags) to be merged into a single text node.

2 This step aligns text nodes into groups so that eventually each group contains the textnodes with the same abstraction (for atomic nodes) or the same set of concepts (for composite nodes).

3 This step aims to divide the values in composite textnodes into individual data units. This step is carry out based on the textnodes in the same group. A group whose values need to be divide is called a composite group.

4 This step is to separate each composite group into multiple aligned groups with each containing the data units of the same concept.

V. MATHEMATICAL MODEL



S1:SOURCE DOCUMENT
S2:SEARCH
S3:ANNOTATION SUPPORT
S4:ONTOLOGY MAINTAINANCE
S5:RDF ANNOTATION
S6:ONTOLOGY
S7:RUN TIME TABLE

S= {s1, s7, Fme, Ff, SHmem}

Where

S1=Input state

S7=Output state
Fme= Friend Function that are s2 to s6
Ff=failure here value null
SHmem=No shared memory use.

Data Unit Similarity

The purpose of data alignment is to put the data units of the same concept into one group so that they can be annotated holistically. In this paper, the similarity between two data units (or two text nodes) d_1 and d_2 is a weighted sum of the similarities of the five features between them, i.e.:

$$Sim(d_1,d_2) = w_1 * SimC(d_1,d2) + w_2 * SimP(d_1,d_2)$$
$$+ W3 * SimD(d_1,d_2) + W4 * SimT(d_1,d_2)\ (1) + w_5 * SimA(d_1,d_2).$$

Tag path similarity *(SimT).* This is the edit distance (EDT) between the tag paths of two data units. The edit distance here refers to the number of insertions and deletions of tags needed to transform one tag path into the other. It can be seen that the maximum number of possible operations needed is the total number of tags in the two tag paths. Let *p1* and *p2* be the tag paths of *d1* and *d2*, respectively, and PLen*(p)* denote the number of tags in tag path *p*, the tag path similarity between *d1* and *d2* is

$$SimT\ (d_1,d_2) = 1 — EDT\ (p1,p2)/PLen(p\{) + PLen(p2)$$

Adjacency similarity (SimA) The adjacency similarity between two data units *d1* and *d2* is the average of the similarity between *d*\ and d^p_2 and the similarity between *d1* and d2, that is

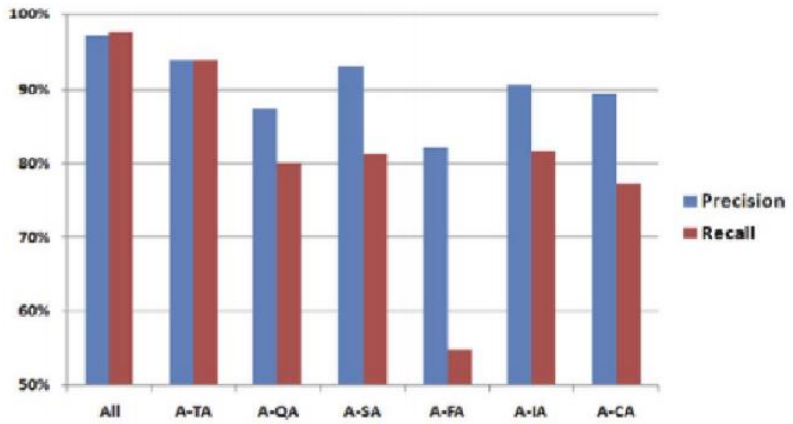
$$SimA(d1,d2) = (Sim'(d^p_1,d_2 2) + Sim'\{d\{,d2))/2.$$

RESULT

In this system we are capable to handling variety of relationship between HTML text node and data units. Our system may result precision and recalls value as follow

Sr No	Domain	Precision	Recall	Performance
1	Book	94	92	Performance of annotation with wrapper
2	Book	96	85	Performance local interface schema

Graphical Result for outcomes using all six type of annotator



CONCLUSION

In this system, we try to implement the data annotation problem and proposed a multi annotator method to automatically constructing an annotation wrapper for annotating the search result records retrieved from any given web database. This method consists of six basic annotators and a way to combine the basic annotators. Each of these annotators exploits one type of features for annotation A special feature of our method is that, when Assign labels the results retrieved from a web database, We also explained how the use of the IIS can help reduce the local interface schema deficiency problem and the inconsistent label problem.

FUTURE SCOPE

1. Searching for semantically and visually similar images on the Web.
2. And mining annotations from them.

ACKNOWLEDGE

First and foremost, I would like to thank my guide, Prof.S.M.Rokade for his guidance and support. I will forever remain grateful for the constant support and guidance extended by guide, in making this system. Through our many discussions, he helped me to form and solidify ideas. The invaluable discussions I had with him, the penetrating questions he has put to me and the constant motivation, has all led to the development of this system.

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, ”Auto-matic Annotation of Data Extracted from Large Web Sites,” Proc. Sixth Int’l Workshop the Web and Databases (WebDB), 2003.

[3] P. Chan and S. Stolfo, ”Experiments on Multistrategy Learn-ing by Meta-Learning,” Proc.SecondInt l Conf. Information and KnowledgeManagement (CIKM), 1993.

[4] W. Bruce Croft, ”Combining Approaches for Information Re-trieval,” Advances in Information Retrieval: Recent Research from the Center for Intelligent Information Retrieval, Kluwer Academic, 2000.

[5] V. Crescenzi, G. Mecca, and P. Merialdo, ”RoadRUNNER: To-wards Automatic Data Extraction from Large Web Sites,” Proc. Very Large Data Bases (VLDB) Conf., 2001.

[6] 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.

[7] H. Elmeleegy, J. Madhavan, and A. Halevy, ”Harvesting Re-lational Tables from Lists on the Web,” Proc. Very Large-Databases (VLDB) Conf., 2009. 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.

[8] D. Freitag, ”Multistrategy Learning for Information Extrac-tion,” Proc. 15th Int l Conf. Machine Learning (ICML), 1998.

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

[10] S. Handschuh, S. Staab, and R. Volz, ”On Deep Annotation,” Proc.12th Int’l Conf. World Wide Web (WWW), 2003. [12] S. Handschuh and S. Staab, ”Authoring and Annotation of Web-Pages in CREAM,” Proc. 11th Int’l Conf. World Wide Web (WWW),2003.

[11] B. He and K. Chang, ”Statistical Schema Matching Across Web Query Interfaces,” Proc. SIGMOD Int’l Conf. Management of Data,2003.

[13] B. He and K. Chang, "Statistical Schema Matching Across Web Query Interfaces," *Proc. SIGMOD Int'l Conf. Management of Data,*2003.

[14] 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.

[15] H. He, W. Meng, C. Yu, and Z. Wu, "Constructing Interface Schemas for Search Interfaces of Web Databases," *Proc. Web Information Systems Eng. (WISE) Conf.,* 2005.

[16] J. Heflin and J. Hendler, "Searching the Web with SHOE," *Proc.AAAI Workshop,* 2000.

[17] L. Kaufman and P. Rousseeuw, *Finding Groups in Data: An Introduction to Cluster Analysis.* John Wiley & Sons, 1990.

[18] N. Krushmerick, D. Weld, and R. Doorenbos, "Wrapper Induction for Information Extraction," *Proc. Int l Joint Conf. ArtificialIntelligence (IJCAI),* 1997.

[19] J. Lee, "Analyses of Multiple Evidence Combination," *Proc. 20th Ann. Int'l ACM SIGIR Conf. Research and Development in Information Retrieval,* 1997.

[20] L. Liu, C. Pu, and W. Han, "XWRAP: An XML-Enabled Wrapper Construction System for Web Information Sources," *Proc. IEEE 16th Int'l Conf. Data Eng. (ICDE),* 2001.

[21] 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.

[22] Y. Lu, H. He, H. Zhao, W. Meng, and C. Yu, "Annotating Structured Data of the Deep Web," *Proc. IEEE 23rd Int'l Conf. Data Eng. (ICDE),* 2007.

[23] J. Madhavan, D. Ko, L. Lot, V. Ganapathy, A. Rasmussen, and A.Y. Halevy, "Google's Deep Web Crawl," *Proc. VLDB Endowment,* vol. 1, no. 2, pp. 1241-1252, 2008.

[24] W. Meng, C. Yu, and K. Liu, "Building Efficient and Effective Metasearch Engines," *ACM Computing Surveys,* vol. 34, no. 1, pp. 48-89, 2002.

[25] S. Mukherjee, I.V. Ramakrishnan, and A. Singh, "Bootstrapping Semantic Annotation for Content-Rich HTML Documents," *Proc.IEEE Int l Conf. Data Eng. (ICDE),* 2005.

[26] B. Popov, A. Kiryakov, A. Kirilov, D. Manov, D. Ognyanoff, and M. Goranov, "KIM - Semantic Annotation Platform," *Proc. Int l Semantic Web Conf. (ISWC),* 2003.

[27] G. Salton and M. McGill, *Introduction to Modern Information Retrieval.* McGraw-Hill, 1983.

[28] W. Su, J. Wang, and F.H. Lochovsky, "ODE: Ontology-Assisted Data Extraction," *ACM Trans. Database Systems,* vol. 34, no. 2, article 12, June 2009.

[29] J. Wang, J. Wen, F. Lochovsky, and W. Ma, "Instance-Based Schema Matching for Web Databases by Domain-Specific Query Probing," *Proc. Very Large Databases (VLDB) Conf.,* 2004.

[30] J. Wang and F.H. Lochovsky, "Data Extraction and Label Assignment for Web Databases," *Proc. 12th Int'l Conf. World Wide Web (WWW),* 2003.

[31] Yiyao Lu, Hai He, Hongkun Zhao, Weiyi Meng, Member, IEEE, *and Clement Yu,* Senior Member, IEEE,2013

