Collaboration of Web Search Result through Automatic Annotation

Jazeb Sayyed
Department of Comp. Engg
DYPCOE
Maharashtra, India

Vikas Mapari
Assistant Professor
Department of Comp. Engg
DYPCOE, Maharashtra, India

ABSTRACT
Web search engines retrieved the large amount of data that is stored in the web database. Internet is the best way to access the data across the world and it present information in user friendly manner. Search engines are designed to retrieved information matching with the user query. When query is submitted web pages are retrieved.Web pages may contain several results set (SRRs).SRR is the collection of data units that represent the real world entity. Now-a-days there is a high demand for extracting and assigning a meaningful label to data units. Many applications like ecommerce and digital libraries required such a system. Therefore an automatic annotation system is used that extracted out data units and aligned them into groups and ensured that each data unit under a group has same semantic with the other data units of the same group. This automatic annotation approach is highly effective and resolves the problem of scalability.

General Terms
Data alignment, Data annotation.SRR

Keywords
Data unit level Annotation, Web database.

1. INTRODUCTION
By considering the demand for retrieving data from multiple web databases, an efficient system is needed to perform searching and retrieval of relevant records without human involvement. The system performs annotation and alignment of data collected by different result pages.Annotation of data is performed for labeling a document. Data annotation enables fast retrieval of information when there is large amount of data is collected [12]. For example, a book comparison shopping system collects multiple SRR from different book selling web sites, it needs to ensure whether any two SRRs refer to the same book or not. The ISBNs can be compared to achieve this. If ISBNs are not available, their titles and authors could be compared. The prices offered by each site are also listed by the system. Thus, the system should aware of the semantic of each data unit. Sample SRR is shown in the figure 1.In this paper, some points are considered like how to automatically assign labels to the data units for each SRRs returned from WDBs. The proposed system is implemented in three phases. Data extraction phase, annotation phase and alignment phase. In data extraction phase different SRRs are collected from different sites.In annotation phase different important features shared among data units, such as their data types (DT),presentation styles (PS), tag path (TP), data contents (DC) and adjacency information (AD) are taken into consideration. There exists basic annotator that helps to figure out the annotation list from extracted data such as table annotator, Query based annotator, Schema based annotator, Frequency based annotator, and Common knowledge based annotator [1].With each of them considering a special type of patterns or features.Six annotators described in short below.

2. BASIC ANNOTATORS
Each returned result page holds multiple SRRs; the dataunits of one concept (attribute) oftenshare some common features. These common features are used to allocate list of annotators.

2.1 Table Annotator (TA)
Each row in table represents an SRR. The table header, which indicates the meaning of each column, is located at the top of the table. Hence table can be used to annotate the SRRs.

2.2 Query Based Annotator (QA)
The query submitted to the search engines is always related to the SRRs of the returned result set. Specifically, the query entered in the search box on the local search interface of the WDB will most likely appear in some retrieved SRRs. Therefore it is considered as annotator.

2.3 Schema Value Annotator (SA)
Attributes on a search interface have predefined values on the interface. For instance, the attribute Publishers may have a set of predefined values (i.e., publishers).The schema value annotator first identifies the attribute that has the highest matching score among all attributes.

2.4 Frequency Based Annotator (FA)
It records the frequency of the data units. The data units with the higher frequency are likely to be attribute names, while the data units with the lower frequency most probably come from databases as embedded values.

2.5 In-Text Infix/Prefix Annotator (IA)
Sometimes 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. Text infix/prefix annotator is used to identity this type of annotators.

2.6 Common Knowledge Annotator (CA)
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” data units in many SRRs from Shopping web sites. Users understand that it is about the availability of the product because this is common knowledge and obvious things [1].

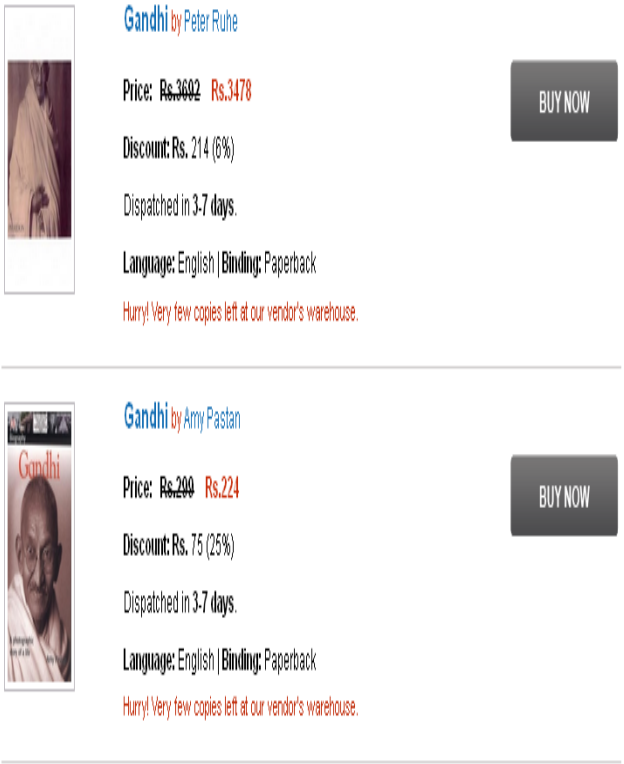

Figure 1: Example of SRR

3. LITERATURE REVIEW

The literature review of the previous work is summarizing below.

3.1 ViDE

Due to complex structures of returned result web pages extracting structured data from deep web pages is a difficult task. Web page programming dependent or precisely HTML document creates some limitations and incapable of handling the ever increasing complexity of HTML source code. This problem is solved by Vision-based Data Extractor. ViDE [2] W. Liu et.al is based on the visual features users can capture on the deep web pages while also utilizing some simple no visual information such as data type and frequent symbols to make the solution more robust.In previous work labeling is performed manually, it is time consuming process and errors can be occurred. Semi automatic annotation is used, in this approach there is no scalability. After that automatic annotation approach came in to exist. Some of the disadvantages of the Vision-based Data Extractor, can only process deep Web Pages contain one data region, while there is number of multi-data region deep Webpage, which is a time consuming process. But visual information of Web pages can help us implement Web data extraction [11]. Mainly Its alignment was only at text node level and not at data unit level.

3.2 ODE

ODE stands for Ontology-Assisted Data Extraction which automatically extracts the results from the HTML web pages. ODE [3] .Su et.al is perform well in determining the query result section ,segmenting the query result section into query result records then it align and label the data values in the query result records. Automatic data extraction is used in meta-querying, data warehousing etc.Here Data extraction is fully automatic and e query result page semantically classified. In Ontology Assisted Data Extraction, in semi automatic annotation there is no additional data is extracted the user need to label only the data in which user is interested. The drawbacks are time consuming; there not used of the large it is not applicable to the large databases websites. To overcome the problems of Roadrunner [4], Dela [1] Yiyao et.al is used that fully automatically extracting the data from the user query result page based on the tag structure that exist on HTML pages.

3.3 RoadRunner

It is the technique for extracting the data from web sites through the use of wrappers that are generated. In this technique it compares the HTML pages and introduces a wrapper based on their similarities and differences. Data is extracted by software modules called wrappers [4] V. Crescenzi et.al. Manually generated wrappers is quite difficult, labor intensive task and difficult to maintain. The main objectives of fully automatic wrapper generation are, Assumption that the wrapper induction system has some knowledge prior to wrapper generation. And one HTML page is examined at a time during the Generation process of a wrapper. Mostly this system was used only for data extraction and not for data annotation.

3.4 ViNsTs

This technique [6] H. Zhao et.al is used for automatic generation of wrappers, used to extract search result record from result page that are generated dynamically. Automatic extraction of search result record is important for many applications. ViNTs [6] H. Zhao et.al utilizes both the HTML tag structure of the source file and visual content features on the result page as displayed on a browser. Visual information And Tag structure based wrapper generator is sufficient for automatically producing wrappers. In this paper main focus is on the issue of how to extract the dynamically generated search result pages returned by search engine. A result page contains multiple SRR's and some of the irrelevant information to the users query. Accurate wrappers completely based on the HTML tag structure. This method makes less sensitive to the misuse of the HTML tags.

3.5 HCRF

HCRF is stands for Hierarchical Conditional Random Field [7] J. Zhu et.al. Existing approaches use to decouple strategies. It attempts to detect the data record and attributes labeling in two separate phases. Separately extract data records and attributes are highly ineffective and propose a probabilistic model to perform both processes simultaneously. HCRF [7] J. Zhu et.al can integrate all useful features by considering their importance, and it can also incorporate hierarchical interaction. It is a template dependent. Main limitation of HCRF is that it's costly as compare to other techniques.

3.6 DeLa

DeLa is used to re constructs web database i.e. It's part of or a "hidden" back-end. It does this by sending queries through HTML forms, automatically generating regular expression wrappers to extract data objects from the result pages and restoring the retrieved data into a table. DeLa performs very good extracting data objects and assigning meaningful labels to the data attributes [8]. The whole process is fully automated, needs no human involvement and proves to be fast and accurate.

4. PROPOSED SYSTEM

The proposed system performs three main steps listed below.

Step 1: Multiple SRRs are extracted out from the from a result pages returned from the web database in response to the user query.

Step 2: In second step data annotation is performed. Several basic annotators is used each exploiting one type of features. Annotators are used to predict a label for the data units within the classified groups and label the data units.

Step 3: In third step data alignment is performed, it aligns all the data into different groups. Each group holds to a different concept. (e.g., all the authors of books are grouped together and all titles of books are grouped together).

The proposed system used automatic annotation solution to implement first two steps. For better classification of the annotator list Naïve Bayes classification algorithm is used in second step. The proposed system is build over considering limitations of previous work. When user submitted its query, data is extracted out from multiple SRRs of multiple websites. We recognized the relationships between text nodes and data unitsthoroughly. Specifically, we identifies four relationship types and provides analysis of each type, there are four basic types (i.e. one-to-one, one-to-many, many to one and one to nothing relationships).For identifying data annotator the naïve bayes algorithm is used to classify the annotators. Main advantages of using naïve bayes are that it is work well with the many complex data also [13]. It required less learning set. It is capable of generating high quality annotation list.

Then alignment algorithm is significantly improved. A new step is added that deal with many-to-one relationship between text nodes and data units (composite text node). With these improvements, the new alignment algorithm takes all four types of relationships into consideration. In improved algorithm we performed clustering of data unit separately. Then we identify near-by data units of the annotator from each SRR and put it at the correct place into our dynamically generated table. The output of the system is dynamically generated table each column head isone of the annotator from annotator list whereas each row represent separate SRR from different sites relevant to query. The system architecture is shown in figure 2.User will submit query to the system. System will collect data units from the different sites. Data extraction phase is performed to get only interested data from the web pages, after that annotation phase is performed that will use naïve bayes classifier to construct annotation list existing system is implemented using clustering algorithm. Multi- annotator approach is used to labeling the classified data. Alignment of data units is performed to place data units at the right place in tabular format.

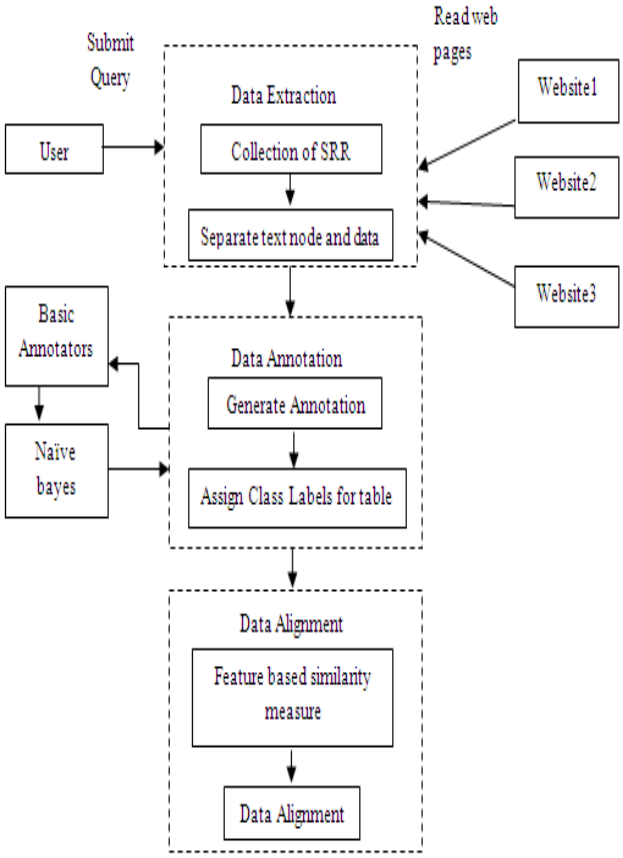

Figure 2: System Architecture

4.1 Data unit Similarity

Data alignment is used to collect data units of the same concept into one group so that they can be annotated holistically [12].The similarity between two data units (or two text nodes) d1 and d2 is a weighted sum of the similarities of the five features between them, i.e.:

$$Sim(d_1, d_2) = w_1 * SimC(d_1, d_2) + w_2 * SimT(d_1, d_2) + w_3 * SimA(d_1, d_2). \quad (1)$$

Weights in the above formula are obtained using a

Genetic algorithm based method [12].

4.1.1 Tag path similarity:

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

$$SimT(d1, d2) = 1 - \frac{EDT(p1, p2)}{PLen(p1) + PLen(p2)} \quad (2)$$

4.1.2 Adjacency similarity:

The adjacency similarity between two data units d1 and d2 is the average of the similarity between dp1and dp2 and the similarity between d1 and d2, that is

$$SimA(d1, d2) = (Sim'(d_{p1}, d_2) + Sim'(ds1, ds2))/2. \quad (3)$$

4.1.3 Data content similarity:

The data content similarity between the d1 and d2 can be

calculated by using following equation

$$SimC(d1, d2) = \frac{V_{d1}.V_{d2}}{||V_{d1}|| * ||V_{d2}||} \quad (4)$$

Alignment algorithm required the similarity between two data unit groups where each group is a collection of data units. The similarity between groups G1 and G2 to be the average of the similarities between every data unit in G1 and every data unit in G2.

4.2 Alignment Algorithm

Following steps describe the working of alignment algorithm in detail.

Step 1: Decorative tags are identified from each SRR and remove it from corresponding SRRs to allow text nodes corresponding to the same attributes that need to be merged into single test node.

Step 2: In second step text nodes are aligned into groups so that each group have text nodes of similar concept for atomic group and same set of concepts for composite groups.

Step 3: In case of one to many relationship between text node and data node. Multiple data unit is encoded into the text node. In third step values in the composite node split into individual data units. For each text node in the group, its text is split into multiple pieces using the separator, each of which becomes element for data unit.

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

5. EXPECTED RESULT

Proposed system will display data units in tabular format by collecting search result from the multiple sites. Data extracted out from the multiple sites.Annotations obtained are accurate as the system uses Naïve Bayes classifier approach, it trains the system to select appropriate annotation. Alignment of data is precise as compare to other methods.

5.1 Performance Measures

For annotation, precision is calculated as the percentage of the correctly annotated data units over all the data units annotated and recall is calculated as the percentage of the data units correctly annotated by the system over all the manually annotated units. The proposed system is implemented using Naive Bayes classifier where as existing system is implemented using Clustering algorithm. For alignment, precision is calculated as the percentage of the correctly aligned data units over all the aligned units by the system and recall is the percentage of the data units that are correctly aligned by the system over manually aligned data units by the expert.

6. CONCLUSION

In this paper, an automatic annotation technique for the Web database underlying for the websites is summarized. The annotation for each data units provides semantic search results for the user. Proposed system is capable of handling a several relationships between HTML text nodes and data units, including one-to one, one-to-many, many-to-one, and one-to-nothing. This approach consists of six basic annotators and all type of annotator makes use of one feature for annotation. In proposed system naïve bayes algorithm is used that generate accurate annotator list from different SRRs. The proposed system takes into consideration composite text node when there are no explicit separators. In short proposed system is used to extract, generate meaningful label automatically and align the data accurately under semantic labels.Automatic Annotating System using Naïve Bayes classifier allows obtaining accurate and meaningful labels.A Bayesian classifier is a probabilistic framework for solving classification problems [13] different classifier can also be used like SVM. For the enhancement of the system image annotation can also be implemented by using this system to display updated information relevant to the user query.

7. ACKNOWLEDGMENTS

I wish to thank and express deep sense of gratitude to our college Dr. D .Y Patil College of Engineering, Ambi and our project guide **Prof. Vikas Mapari** for his consistent guidance, inspiration and sympathetic attitude throughout the project work. This project paper work would not have been feasible without encouragement and guidance of our PG coordinator, **Prof.Anupkumar Bongale.** I am grateful to **Prof. Sandeep Kadam,** Head of Computer Department, and DYPCOE for always being ready to help with the most diverse problems that I have encountered along the way.

8. REFERENCES

[1] Yiyao Lu, Hai He, Hongkun Zhao, WeiyiMeng "Annotating Search Results from Web Databases "IEEE Transaction on Knowledge and Data Engineering, Vol. 25, No.3, March 2013.

[2] 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.

[3] 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.

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

[5] 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.

[6] H.Zhao, W. Meng, Z.Wu, V.Raghavan, and C.Yu,"Fully Automatic Wrapper Generation for Search Engines," Proc. Int'I Conf. World Wide Web, 2005.

[7] J. Zhu, Z. Nie, J. Wen, B.Zhang, and W-Y.Ma," Simultaneous Record Detection and Attribute Labeling in Web Data Extraction,"Proc. ACM SIGKDD Int'I Conf. Knowledge Discovery and Data Mining, 2006.

[8] B. Adelberg. "NoDoSE - A tool for semi-automatically extracting structured and semi structured data from text documents," Proc. ACM SIGMOD Conf., 1998, 283-294.

[9] R. Baumgartner, S. Flesca and G. Gottlob."Visual web information extraction with Lixto," Proc.27th VLDB Conf., 2001, 119-128.

[10] 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)*