Generating Data Annotation for Web Records

[1] Ms. Priyanka Ashok Jadhav ,[2]Ms. Snehal Nargundi

[1]Department of Information Technology ,[2] Department of Information Technology
[1]RMD Sinhgad School of Engineering,Pune,India ,[2] RMD Sinhgad School of Engineering,Pune,India

Abstract – **In our day to day life web search has become integral part of many people life. This web search is database based. When user searches something on web, the portion of web is accessible to user through HTML based interface. For each query submitted to search engine, data instances returned from database are encoded in result pages dynamically for human browsing. Encoded data unit which need to be machine processable should be extracted and assign meaningful labels to it. In this paper I am presenting an dynamic data annotation approach which first align data unit on result pages into different groups such that each group will have same semantic. These group are annotated from different aspect and aggregate the different annotation to predict final annotation label for it. After annotation user can also group or order the content according to their wish as well as check the popularity of content.**

Index Terms – **Data alignment, data annotation, web database**

I. INTRODUCTION

Large portion of web is database based. Many search engine store multiple or number of web pages. Data encoded in the returned result pages of many search engines come from the underlying structured databases such as relational databases. Such type of search engine is called as *Web Database* (WDB). Result pages returned from web database are referred as **s**earch **r**esult **r**ecord (SRR). Each search result record corresponds to real world entity and each SRR has multiple data units. For example in Fig 1.book title, publisher, author, price. Not all data unit are encoded with meaningful label. In first line of SRR are not labeled with “ Title ” even though human user can recognize it easily. This paper address how dynamically annotate data unit in SRR returned by WDB and assigning meaningful label to them. Due to rapid growth of web database annotation problem has become very significant. In early days application require tremendous human effort to annotate data unit manually. For given set of SRRs that are extracted from result page returned from WDB. An automatic annotation solution consists of three phases. Phase 1 is called *alignment phase*. In this phase we identify all the data unit in SRR and organize them into different group such that each group will have different concept across all SRR. Phase 2 is called *annotation phase*. In this phase table annotator is used to produce label for data units within their group. Last phase is call *annotation wrapper*, for each identified concept annotation rule are described to determine how to extract data unit concept in result page.

II. RELATED WORK

Annotating structured data from large web database is relevant to information extraction. This section is brief review of annotation task. Many systems rely on human user to rank the specific information on sample pages and label ranked data item at the same time. Then the rules are used to extract the set of information on web pages. These systems are often referred as wrapper introduction system [9][10]. But it suffers from poor scalability and not suitable for application that need to extract information from large number of web sources.

Several works are done to automatically assigning meaningful label to the data unit in SRR [4]. Data unit with close label are annotated on result pages. In ODF[3] the use of query interface and result pages from WDS, ontologies are constructed in some domain. The label are assigned to each data unit using domain ontology. After labeling, data values having same label are get aligned. Dela [8] uses HTML tags to align data unit. It fill the data unit into the table through regular expression based data tree algorithm.

ViDIE [2] uses data record extraction and item extraction technique. It uses visual features such as position feature, layout feature, content feature, appearance feature etc. to perform alignment and generate alignment wrapper. But this alignment is at only text node level not data unit level. WISE [7] automatically integrate local interface into global interface using only domain independent knowledge.

In this paper, data alignment approach is different from previous work in following aspect. Firstly this approach handles all type of relationship between text node and data unit, while existing approach use only some type such as one to one and one to many relationship. Second this approach use different features together while existing approach use only some features such as HTML tag [8] and visual feature [5]. Last i.e. third, I am using clustering algorithm to perform alignment.

This paper is an extension of previous work [1][4]. Following are improvement over [1][4] paper. This paper identifies four type of relationship and provides analysis of each type. Secondly alignment algorithm is significantly improved it also handle many to one relationship between text node and relationship between text node and data unit. Clustering algorithm is used to handle one to nothing relationship. This paper also adds machine learning technique called *Standford Parser*. This parser take raw text as input and give the base form of word, their parts of speech, whether they are names of people, companies etc. normalize data etc. With automatic data annotation this paper also add option for user to order or group the data according to their wish. It also display the popularity of product among the list of annotated data.

III. SYSTEM ARCHITECTURE

Fig. 4 shows the system architecture. System architecture describes annotation process. Detailed approach of architecture is as follows.

Web Crawling

Input to the system is SRR extracted from the result pages by search site. These records are extracted by crawling the web. Data is extracted from different domain such as book, printer, mobile, laptop.

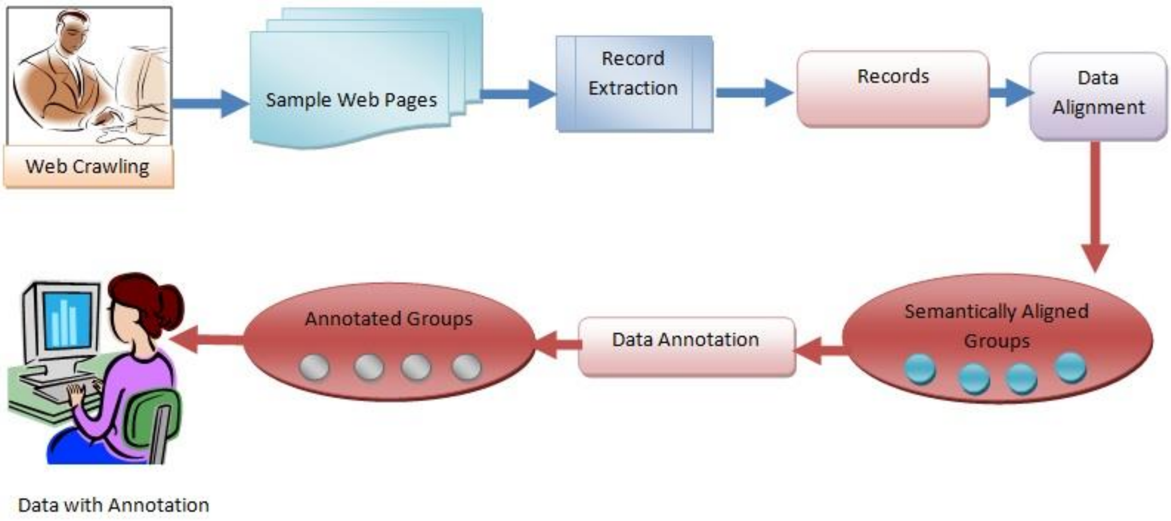


Figure 4. System Architecture

Record Extraction

For annotation SRRs need to be extracted from the result pages returned from search site. SRR must be correctly extracted by discarding irrelevant information such as advertisement and sponsored links from each result page. ViNT system [6] is used here to extract the SRRs from search engine returned result pages. SRRs retrieved from result pages are visually arranged together but obviously separated and they have similar shape, content and positional features. ViNT uses both the visual features and HTML tag structures of the result pages. ViNT is fully auto- mated and domain independent.

Data Alignment

Data alignment is performed to put data unit of some concept into one group. As each extracted SRR typically contain multiple ordered node in HTML tag structure of result page. A node may contain a single data unit or multiple data units. When SRRs are extracted, the corresponding data units are not aligned together. Data alignment use data units from different SRRs and group them together to form semantically separate group. Data alignment is done based upon the assumption that data unit belonging to the same semantic or concept in different SRR often share similar fixed layout and presentation style across all SRR and they appear in same order across all SRR on the same result page. If we consider the SRR on the result page in table format, where each row represent one SRR and each cell in every row contain a data unit.

Aligned Group

When designing a web search clustering algorithm, special attention must be paid to ensuring that both content and description (labels) of the resulting groups are meaningful to humans. Lingo clustering algorithm[11] is used here for creating semantic cluster. Pseudo code of lingo algorithm is as follows.

Pseudo-code of the Lingo algorithm

```
D   input documents (or snippets)
        {STEP 1: Preprocessing}
for all d 2 D do
perform text segmentation of d; {Detect word boundaries etc.}
if language of d recognized then
apply stemming and mark stop-words in d;
end if
end for
        {STEP 2: Frequent Phrase Extraction}
concatenate all documents;
Pc ← discover complete phrases;
Pf ← p : {p ∈ Pc ^ frequency(p) > Term Frequency Threshold};
        {STEP 3: Cluster Label Induction}
A ← term-document matrix of terms not marked as stop-words and
```

```
        with frequency higher than the Term Frequency Threshold;
$\sum$,U, V ←SVD(A); {Product of SVD decomposition of A}
k ← 0; {Start with zero clusters}
n ← rank(A);
repeat
k ← k + 1;
q ← $(\sum_{i=1}^{k} \sum ii)/ \sum_{i=1}^{n} \sum ii$
until q < Candidate Label Threshold;
P ← phrase matrix for Pf ;
for all columns of UT
  k P do
find the largest component mi in the column;
add the corresponding phrase to the Cluster Label Candidates set;
label Score ← mi;
end for
calculate cosine similarities between all pairs of candidate labels;
identify groups of labels that exceed the Label Similarity Threshold;
for all groups of similar labels do
select one label with the highest score;
end for
        {STEP 4: Cluster Content Discovery}
for all $L \in$ Cluster Label Candidates do
create cluster C described with L;
add to C all documents whose similarity
to C exceeds the Snippet Assignment Threshold;
end for
put all unassigned documents in the “Others” group;
        {STEP 5: Final Cluster Formation}
for all clusters do
cluster Score ← label Score × ||C||;
end for
```

Step 1 Preprocessing

Stemming and stop words removal are very common operations in Information Retrieval. Interestingly, their influence on results is not always positive in certain applications stemming yielded no improvement to overall quality The aim of the preprocessing phase is to prune from the input all characters and terms that can possibly affect the quality of group descriptions. Three steps are performed: text filtering removes HTML tags, entities and non-letter characters except for sentence boundaries. Next, each snippet’s language is identified and finally appropriate stemming and stop words removal end the preprocessing phase.

Step 2 Frequent phrase extraction

frequent phrases defined as as recurring ordered sequences of terms appearing in the input documents. Intuitively, when writing about something, we usually repeat the subject-related keywords to keep a reader’s attention. Obviously, in a good writing style it is common to use synonymy and pronouns and thus avoid annoying repetition. A The complete phrase is a complete substring of the collated text of the input documents. To be a candidate for a cluster label, a frequent phrase or a single term must:

1. appear in the input documents at least certain number of times (term frequency threshold),
2. not cross sentence boundaries,
3. be a complete phrase (see definition below),
4. not begin nor end with a stop word.

Step 3 Cluster label induction

Once frequent phrases (and single frequent terms) that exceed term frequency thresholds are known, they are used for cluster label induction. There are three steps to this: term-document matrix building, abstract concept discovery, phrase matching and label pruning.

The term-document matrix is constructed out of single terms that exceed a predefined term frequency threshold. Weight of each term is calculated using the standard term frequency, inverse document frequency (tfidf) formula terms appearing in document titles are additionally scaled by a constant factor. In abstract concept discovery, Singular Value Decomposition method is applied to the term-document matrix to find its orthogonal basis. Phrase matching and label pruning step, where group descriptions are discovered, relies on an important observation that both abstract concepts and frequent phrases are expressed in the same vector space

Step 4 Cluster content discovery

In the cluster content discovery phase, the classic Vector Space Model is used to assign the input documents to the cluster labels induced in the previous phase.

Step 5 Final cluster formation

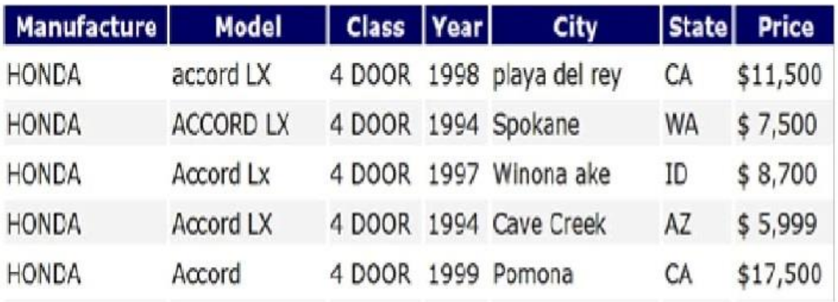
Finally, clusters are sorted for display based on their score, calculated using the following simple formula:
Cscore = label score × ||C||, where ||C|| is the number of documents assigned to cluster C. The scoring function, although simple, prefers well-described and relatively large groups over smaller, possibly noisy ones.

Data Annotation

Once we aligned group we need to annotate these groups. This section is brief description of different type of annotators to annotate aligned groups.

A. Table Annotator

Many WDBs use table to organize returned SRRs. Each table will multiple rows and columns and each row represents SRR. Table header is used to indicate meaning of each column, which is located at the top of the table. Data unit with same semantic are aligned with its corresponding column header. Fig 6. shows SRR in table format. Working of table annotator is as follows. First it identifies all column header of table, second for each SRR data unit in every cell are taken and column header whose area has maximum overlap with the cell are selected. These data unit are the assigned with this column header and then labeled by the header text *A.* The remaining data unit are processed similarly.



Manufacture	Model	Class	Year	City	State	Price
HONDA	accord LX	4 DOOR	1998	playa del rey	CA	$11,500
HONDA	ACCORD LX	4 DOOR	1994	Spokane	WA	$ 7,500
HONDA	Accord Lx	4 DOOR	1997	Winona ake	ID	$ 8,700
HONDA	Accord LX	4 DOOR	1994	Cave Creek	AZ	$ 5,999
HONDA	Accord	4 DOOR	1999	Pomona	CA	$17,500

Figure 6. SRR in table format

B. Query Based Annotator

SRR returned from WDBs are always related to specific query. Specifically, the query terms specified on some attributes in the interface of the web database are most appear in some retrieved SRRs. For example, if a query term (say "database") is submitted using "Title" on a book search interface, then the titles of the returned book records will likely contain "database". Thus, the "Title" attribute can be used to annotate the title values of these book records. Working of query bases annotator is as follows: Given query is submitted against attribute k on the local search interface, the it first find the group that has maximum number of occurrences of query terms and then assign *gn(k)* as table group.

C. Frequency Based Annotator

In Fig. 1 "Our Prize" appear in three record and followed prize value are different in these records. In this type data unit with higher frequency are likely to be attribute name as part of template program for generating records while data unit with low frequency comes from database embedded values. Let G_i be the group whose data units are not the same (indicating a lower frequency). Frequency-based annotator aims to find common preceding units. These data units are shared by all the data units from group G_i. All found preceding data units are concatenated to form the label for the group G_i. For example Consider Fig.1 In the data alignment phase, a group is created for $17.50, $18.95 and $20.50. The data units from this group have different values. These three values share the same preceding unit "Our Price", which occurs in all SRRs. "Our Price" don't have any preceding data units . Reason for this is, this data unit is the first unit in the same line. Hence, the frequency based annotator will annotate every above prize with the label "Our Price".

IV. RESULT ANALYSIS

We have data from book, printer ,mobile and laptop domain and clustering performed on these data. Result Analysis is done by comparing the values of cluster and experimental values. Precession and recall (specificity) measure of information retrieval are used here to calculate performance of our measure . Precession is percent of correctly aligned data unit by all aligned unit by system and recall is percent of data unit that are correctly aligned by system over all manually aligned data unit. Data unit is said to be correct if system assigned label has same meaning as its manually assigned label. Fig. 7 shows precession graph and fig 8. shows specificity graph.

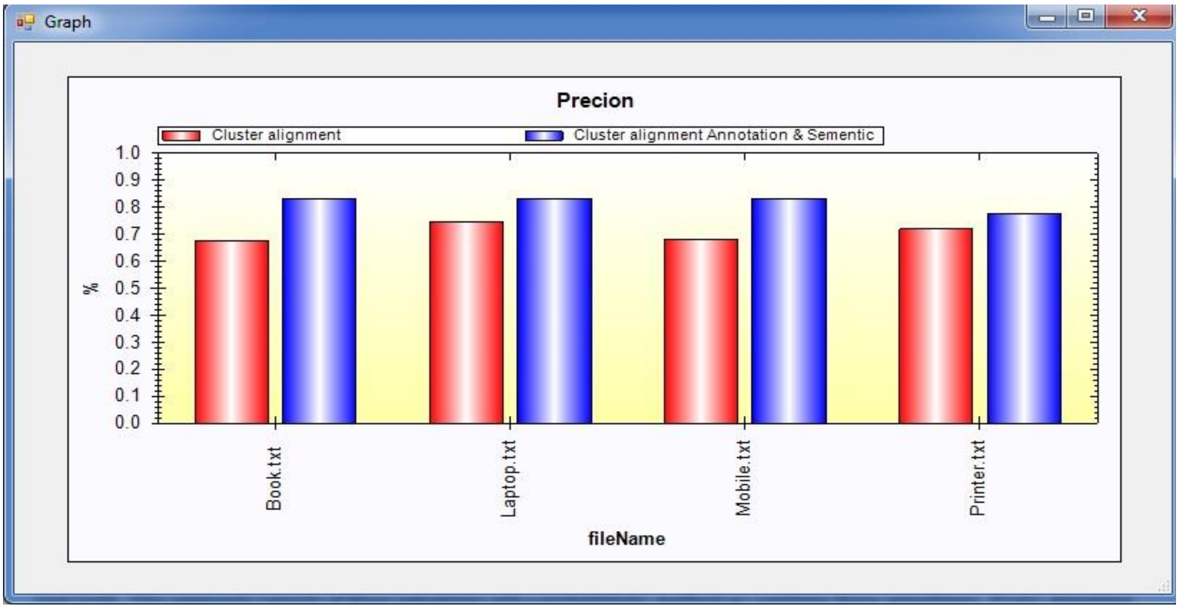


Figure 7 Result of precession

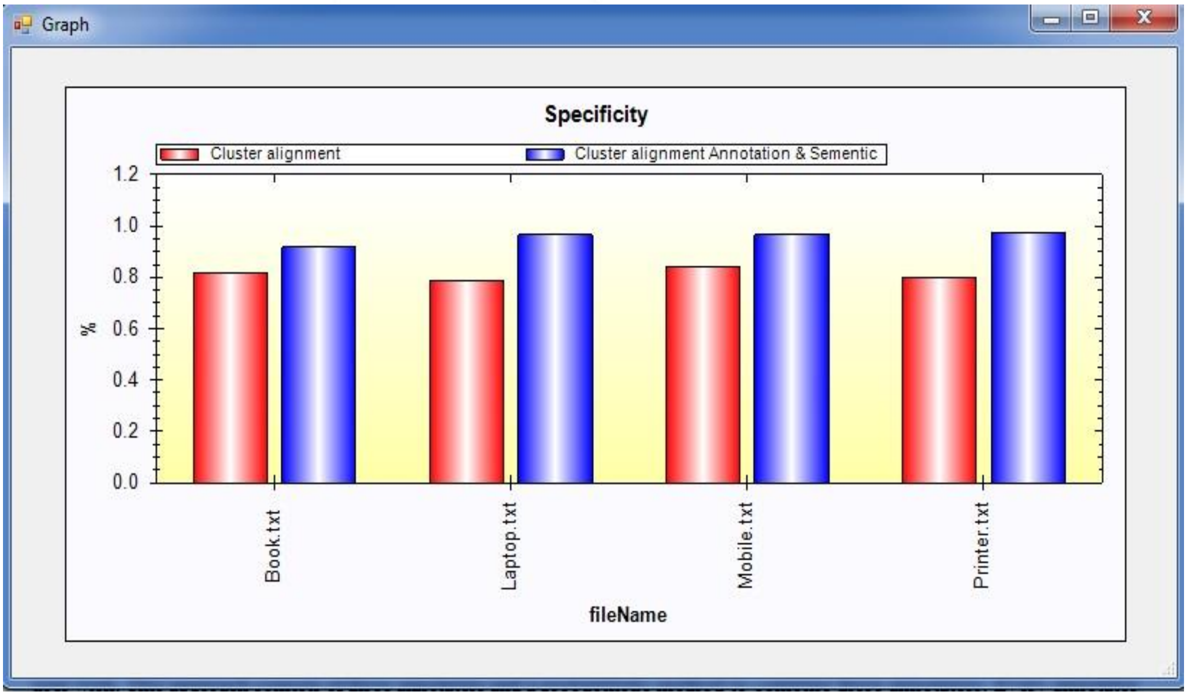


Figure 8 Result of Recall

V. CONCLUSION

This paper, explains data annotation and proposes a multi annotator approach to automatically annotate data unit according to user wish. This approach consists of three annotators and a probabilistic method to combine these annotators. Every annotator explains one type of features for annotation. This paper also explains automatic data alignment problem. Its critical to achieving accurate annotation. Clustering based shifting method handles variety of relationships between HTML text nodes and data units, which includes one-to-one, one-to-many, many-to-one, and one-to-nothing. Standford parser is used to achieve more accurate result. Data unit with different label and same semantic are detected and parsed by this parser and select which label should be useful for labeling data unit.

VI. ACKNOWLEDGMENT

First and foremost, I would like to express my sincere gratitude to my guide Ms. Snehal Nargundi and Ms. Sweta Kale for their continuous support .I will always be grateful for their patience, motivation, enthusiasm, and immense knowledge. This work is an extension of the previous research efforts of Yiyao Lu, Hai He, Hongkun Zhao, WeiyiMeng. I am very much indebted to them for their inspiring work which is a boost for future researchers. Finally, I would like to pay my respect and love to my parents and

husband for their encouragement throughout my career.

References

[1] Y. Lu, Hai He, H. Zhao, Weiyi Meng. "Annotating Search Result from Web Database" IEEE transactions 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] 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.

[5] 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.

[6] 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.

[7] 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.

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

[9] 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.

[10] N. Krushmerick, D. Weld, and R. Doorenbos, "Wrapper Induction for Information Extraction," Proc. Int'l Joint Conf. Artificial Intelligence (IJCAI), 1999.

[11] Stanis law Osi´nski, Jerzy Stefanowski, and Dawid Weiss "Lingo: Search Results Clustering Algorithm Based on Singular Value Decomposition"