ISSN:0975-9646

Web Database Annotation for Fast and Accurate Retrieval

VijayaLakshmi K,

M.Tech, CSE Department,
Madanapalle Institute of Technology & Science,
JNTUA, Madanapalle, AP, India.

Sudhakar Yadav N

Asst. Professor, CSE Department,
Madanapalle Institute of Technology & Science,
JNTUA, Madanapalle, AP, India.

Abstract -- Mounting of databases have become web accessible through Hyper Text Mark up Language based explore interfaces. When people browse for particular purpose the search results are obtained relevant to data or irrelevant to data. To get the desired data people need to search again from the obtained search results. To overcome this problem, we introduce an automatic marginal note method that initially arranges the data units on a resultant page of a search site into different groups. Then, for each and every group we build marginal notes from various aspects and combine the distinct labels to estimate an absolute label for it. Lastly marginal note binder for each site is dynamically constructed and used for making annotations automatically when we search newly from the same web DB.

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

I. INTRODUCTION

Huge volumes of data are maintained in the data bases for future use. Data mining is the process of extracting hidden knowledge from large volumes of raw data. The efficiency of data searching and updating of information is achieved by alignment and annotation. Annotation allows accumulation of information to a file, paragraph or phrase. In other words assigning meaningful labels to data is called data annotation.

Web is a pool of information for which data annotation enables quick retrieval from web. When people browse for particular purpose the search result records obtained relevant to data or irrelevant to data, each such search result has multiple data units. Data units are real world entities which are dynamically encoded into result pages for human surfing purpose but later transformed into machine process able unit and allotted significant labels. Manual encoding of data units needs lots of man power which lack in scalability. For this reason an automatic annotation method is required to assign data units within the search result records automatically.

This approach first arranges the data units into different groups and ensures that each data unit within a group that have semantic meaning. Each group is then labelled considering several aspects and combined to forecast an absolute label. At last, a wrapper is build. Wrappers are generally used as translators which label new response pages from the same web database. This automatic annotation approach is very proficient and scalable.

II. RELATED WORK

Existing System:

The labelling of data units in search result records are automatically assigned labels that are returned from WDBs. It includes relationships between data units and text nodes; they are many to one and one to nothing.

Disadvantages

1. It is critical to achieve alignment holistically and accurate annotation.
2. Splitting of composite text nodes is possible only when explicit separators are present

III. PROPOSED WORK

We are extending this work to improve our technique in order to divide composite text node when explicit separations are not present. We also try to include various machine learning techniques and site with more sample pages to recognize the best technique for the data alignment difficulty.

Advantages of proposed system

1. Works more efficiently for data alignment compared to previous work.
2. Improve our technique in order to split composite text nodes when explicit separations are not present.

System Architecture

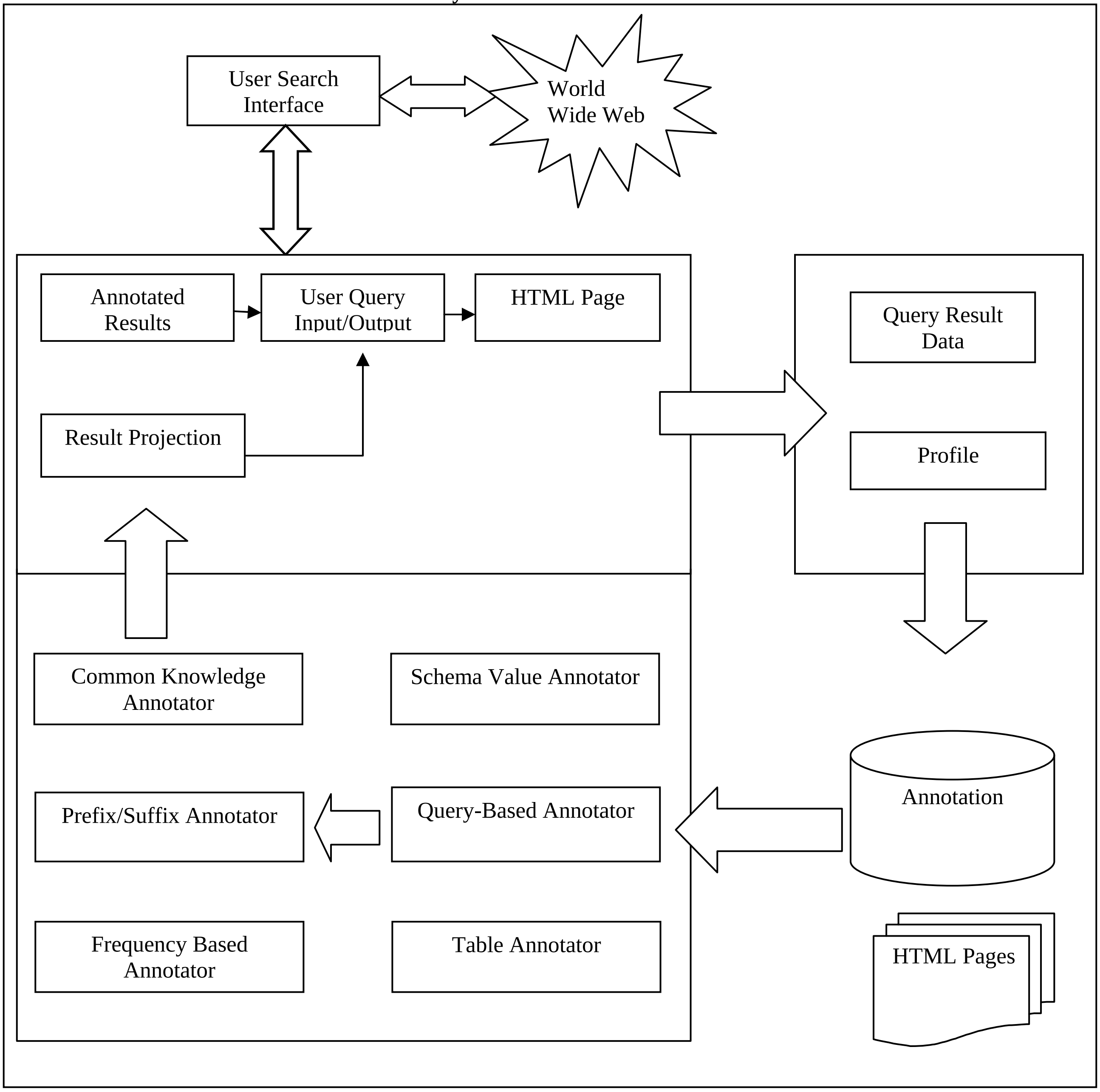

Fig: Architecture of web database annotation for fast and accurate retrieval

Automatic Marginal Note Method:

A composite text contains data units and their data types. Consider, composite a text node "Yardley/2000/1931618416/0.05557" and its data types are given as <First-Letter-Uppercase-String> <Symbol> <Int> <Symbol> <Int> <Symbol> <Decimal>. Successive similar data type terms are considered as one term and only one among them is kept. Every data type apart from Ordinary String has definite patterns so that they can be simply recognized. The data units of the similar model having the similar set of concepts generally have the similar data type.

Step 1: Load the website in the system

Step 2: Retrieve the articles from it using various links available on the pages of the website

Step 3: Look for heading tags, bold/strong tags, and frequency of the words in the articles to decide the annotation for the specific articles

Step 4: Maintain the list of the various articles and provide them as quick link lists for future usage

Step 5: Update the annotation list using every new website visited using the above steps

Step 6: then each data unit is assigned to a cluster, such that N clusters have N data units. Let the distances between the clusters is same as the distances between the data units they hold.

Step 7: The most similar pair of clusters are identified and merged into a single cluster, so that now you have one cluster less with the help of Document Frequency

- The weights in the cosine-similarity are found from the TF-IDF measure between the phrases (meta-tags) of the two documents.
- This is done by computing the term weights involved.
- TF = C / T
- IDF = D / DF.

D → quotient of the total number of documents
DF → number of times each word is found in the entire corpus
C → quotient of no of times a word appears in each document
T → total number of words in the document
TFIDF = TF * IDF

Step 8: Compute distances between the new cluster and each of the old clusters.

Step 9: Repeat above until all data units are clustered into a single cluster of size N.

A. Performance Evaluation:

Performance Using Local Interface Schema:

We conducted experiments on the basis of WDBs chosen from the areas: retail, movie, music, sports, health care, electronics, and education. Any WDB, search interfaces in general contain a few attributes of the underlying data. When a query is presented to a search interface, the response pages also contain definite unseen schema. The resultant schema and the LIS generally distribute a large number of attributes. Some attributes of the underling DB are not appropriate for identifying query constraints as established by the developer of the WDB, which are not included in LIS called LIS inadequacy further leads to inconsistent labeling problem. That is distinct labels are allocated to similar data units retrieved from various WDBs since distinct LISs assigns distinct names to semantic attributes. When we run the annotation process using LIS built for each WDB, we observe LIS has comparatively little impact on precision but large effect on recall because of LIS adequacy as shown in fig (a).

Domain	Precision	Recall
Retail	98.70%	98.80%
Education	96.80%	95.40%
Electronics	97.30%	97.30%
Sports	95.10%	95.10%
Health care	96.80%	100.00%
Movie	96.80%	97.70%
Music	98.30%	98.70%
Avg	97.11%	97.57%

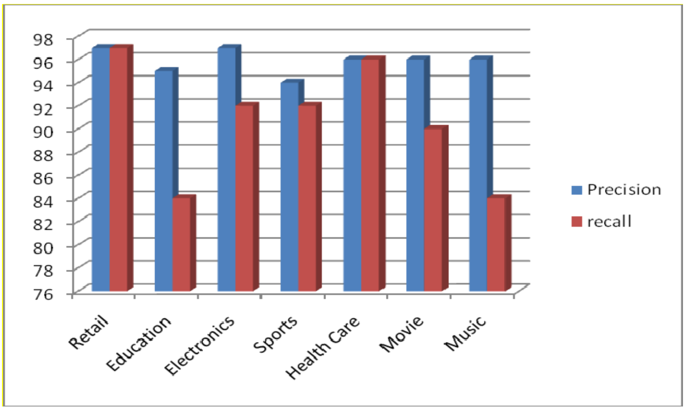

Fig (a): Performance Using LIS

Performance Using Integrated Interface Schema:

This method uses a wise integrator approach to construct an IIS over many WDBs in that domain that has been used. This IIS integrates the attributes of LISs. All the corresponding attributes from various LISs and their values in the local interfaces are merged as the values of the integrated global attributes, every global attribute has only one global name and an attribute mapping table is build to set up mapping between the name of each LIS attribute and its corresponding name present in IIS. The metrics used to measure the performance of LIS and IIS are precision and recall. Precision is the fraction of the appropriately aligned data units on the whole system aligned data units. Recall is the fraction of the data units that are appropriately aligned by the system on the whole manually aligned data units by the specialist. The IIS can indeed increase the annotation performance that are shown in the below fig (b).

Domain	Precision	Recall
Retail	97.30%	97.30%
Education	95.10%	84.60%
Electronics	97.50%	92.70%
Sports	94.90%	92.20%
Health care	96.30%	96.10%
Movie	96.80%	90.80%
Music	96.20%	84.90%
Avg	95.40%	90.10%

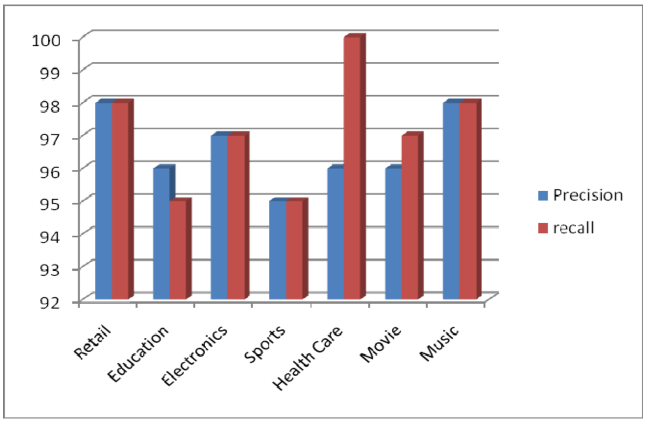

Fig (b): Performance Using IIS

IV. Conclusion

The difficulty in data annotation is analysed and anticipated a multi-annotator approach has been proposed for dynamically constructing marginal notes for the search result records retrieved from any given web database. The six basic annotators provide high quality marginal notes by exploiting one type of feature for annotation. This approach uses both LIS & IIS of web databases in the same domain to avoid LIS difficulties of insufficiency and the incompatible labelling. This technique is also capable of managing a diversity of relationships between HTML text nodes and data units including one-to-one, one-to-many, many-to-one, and one-to-nothing.

Scope of future enhancement:

Future work, we will investigate methods to exploit in order to divide composite text node when no explicit separators are present and try to use various machine learning techniques, use additional sample pages from each training site to get the future weights to recognize the finest technique to the data alignment crisis.

References

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

[2] V. Crescenzi, G. Mecca, and P. Merialdo, “RoadRUNNER:TowardsAutomatic Data Extraction from Large Web Sites,” Proc. Very Large Data Bases (VLDB) Conf., 2001.

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

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