Enhancing Automatic Extraction of Top-K List from Web

Dipali S. Patil[1]
Prof. N. A. Dhawas[2]
STES's Sinhgad Institute of Technology, Lonavala.[1,2]

Abstract- **Now a day's World Wide Web is considered as biggest resource of information. This large database which contains information in all area but finding particular information or extracting accurate data from web is difficult. The strong reason behind this sentence is that the data available on this huge database is not in same format. When data is in particular format you can extract information without any difficulty when extract data from HTML pages, we select data easily with the help of tags. This paper is extracting top-k list from all available web database which contain data either in structured or unstructured format. An algorithm is implemented for this reason which provides an accurate and faster generation of top-k list.**

Keywords:- *Classifier, Content Processor, Database ,Parser, Top-k list.*

I. INTRODUCTION

The World Wide Web is biggest resource of information. The information available on the World Wide Web is either in structured or semi structured form. World Wide Web contains information in natural language format. The information available in table format is structured or semi structured information because it contains tags (HTML pages).Acquiring useful information from this structured form or semi structured form is easy and lot of work had done on this. But collecting specific information from such non-structured data is very difficult and time consuming task.

Extracting useful information from huge resource is called data mining and if that huge resource is web then it is called as web mining. In this paper we propose a new approach to retrieve top-k list information from web pages (which are in structured or unstructured format). We are using top-k list information eg."Top 10 software companies in the world". We are using top-k list because it contains huge information with proper semantics. We are developing new method to retrieve top-k list from web pages with high accuracy and it will help to retrieve information from unstructured web database. Our algorithm is work on number of k items with topics and ranking criteria, we also provide some optional parameter like time and location to retrieve proper information from the web list .This application is tested on large web database with good precision and recall.

The research on the Top-k list extraction was already done, it extracts structured or semi structured data from web [3]. That data must be in HTML web pages. This task is easy because with the help of HTML tags data is selected. Take one simple example: tabular information is present in HTML format, which can separate the data with the help of tag. Same tag contains similar type of information.

Our system works on Top-k list. This is a list with proper and high quality data. Top-k list example is: "Paper 2014: Top 10 criteria for selecting paper". This sentence we can divide in three segments. First segment is "Paper 2014: top 10", segment two is "criteria" and last segment is "for selecting paper". In above example Paper and 2014 are classifier, top is criteria, 10 is k, criteria is concept and remaining part is splitter. With the help of this we can find more specific data list.

This paper is providing extraction algorithm which is designed for extraction top-k list and also give system data flow details. In the implementation of this algorithm we are using parser, web page title classifier, top-k list candidate picker, the ranker for top-k list and content processor. Input for our system is web page in any language. That page is provided to parser as input then that information is forwarded to title classifier which selects pages whose titles are matches with given top-k list. Other pages are discarded. Selected pages are forwarded to candidate picker. It can select data from pages from page as candidate list. After this top-k ranker can rank that candidates. This top-k list is processed by content processor and we can get final and accurate output i.e. top-k list.

The second section of this paper contains overview of previous systems. This section presents summary of other related techniques. Section III contains detail information regarding our new proposed system. This section contains data flow diagram of system and proposed algorithm in detail. Last section contains future scope for proposed system.

978-1-4799-3759-2/14/$31.00©2014 IEEE

II. LITERATURE SURVEY

Table 1.Literature Review

Rf. No.	Year	Concept	Advantages	Disadvantages
[1]	2013	• Retrieving top-k list data From different form of web pages	• Retrieve accurate information • Extract Top-K list	Only applicable to HTML pages because HTML Parser is used
[2]	2011	• Hybrid approach for discovering general list and extracting it from web.	• Gives efficient and fast result.	It gives general list not ranked list.
[3]	2009	• Mines contagious and non-contagious data records	• Able to discover non-contiguous data records	Less Accuracy
[4]	2008	• Extract tabular form information from HTML pages	• Used to Extract tabular information	Only applicable to Tabular information

It projected a novel and helpful technique to without human intervention mine data records from web page [4]. Uses algorithm MDR (Mining Data Records in Web Pages) Algorithm finds all data records by using related tags. This method of extraction is based on two things:

1. Data records that describes similar objects that resides in neighboring area of webpage and tagged by similar HTML tags. Though we got that region the problem is that the every data record is having different length because it provides different data.
2. All similar records are having same HTML tag. When we are forming tree all that records is come under the same parent. This indicates our all data records are reside in same memory area.

The proposed system in this paper is working in 3 steps:

a. At start we are preparing tag tree of HTML pages
b. We mine data region first.
c. After getting data region we can search data records into that region.

In previous paper they find only that records which are in similar data area. But this paper invents technique in which they can search and detect similar data records which are present outside of one data area.

The paper provides solution for the extracting structured data. That means it give you solution to extract structured data from web which is available in tabular format. That described the Web Tables system, which was the first large-scale attempt to extract and leverage the relational information embedded in HTML tables on the Web In this paper they are given an algorithm which uses ACSDb (Attribute Correlation Statistics Database) and provide a auto complete tool to help database designers to choose a scheme.

They applied the Attribute correlation statistics database (ACSDb) to a number of schema-related problems: to improve relation ranking, to construct a schema auto-complete tool, to create synonyms for schema matching use, and to help users in navigating the ACSDb itself [3].

In last section of the paper author describes the applications of the ACSDb. In that first is the schema auto complete, when user enters one or more domain related attributes, the schema auto completer guesses the rest attribute labs and they are appropriate to target domain. Next application is Attribute Synonym-Finding. In schema matching the schema matchers have used a synonym set from handmade thesaurus. By using this application schema matcher can automatically find synonyms between arbitrary attribute strings. The last application is Join Graph Traversal.

[1]This paper presents a novel and interesting problem of extracting top-k lists from the web. There is large scale availability of top-k list data on web so that this paper is focusing on top-k list. We can rank this top-k list data. Also it has interesting semantics. For extraction of top-k list, here we recognize top-k pages, then extract top-k list and at the last contents of the list are understand. Authors used tag path clustering in this paper, which has specific goals, it uses numeric k and considering list as instances with respect to the concept in the title and it is time saving process. But it has disadvantage that it accepts HTML pages only [4]. They demonstrated an algorithm that automatically extracts over 1.7 million such lists from the web snapshot and also discovers the structure of each list. It will process only those

web pages which are available in HTML language because it contains HTML parser. Their evaluation results show that the algorithm achieves 92.0% precision and 72.3% recall [1].

The paper "A System for Extracting Top-K List from the Web" is published in 2012 by the authors Zhixian Zhang, Kenny Zhu, Haixun Wang. In this paper they are extracting top-k list from web pages which are in HTML form only (structured or semi structured format). This system carried out work in three steps:

1. In first step it recognize top-k list [2].
2. At this step it picks up particular contents of top-k list.
3. Here at the last it understands the list plus process that list.

In this paper they are providing system overview also which contains information regarding system flow. That will contain:

1. Title classifier
2. Candidate picker
3. Top-k ranker information

III. PROPOSED SYSTEM

The title of a top-k page contains at least three pieces of important information: i) A number k, for example, 20, Twelve, and 10 in the above example, which indicates how many items are described in the page; ii) A topic or concept the items belong to, for example, Scientists, Children's Books, Hollywood Classics and podcasts; iii) A ranking criterion, for example, Influential, Interesting, and You Shouldn't Miss (which is equivalent to Best or Top).

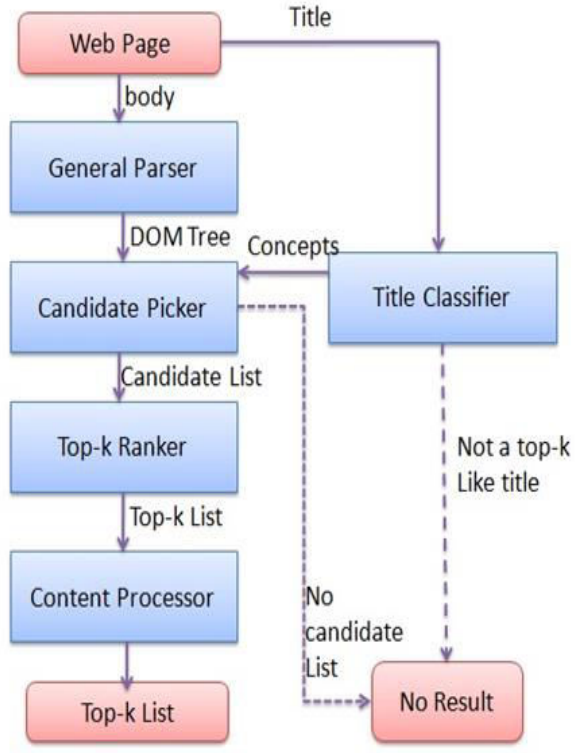

Fig.1. Proposed System

The title of a top-k page contains at least three pieces of important information: i) A number k, for example, 20, Twelve, and 10 in the above example, which indicates how many items are described in the page; ii) A topic or concept the items belong to, for example, Scientists, Children's Books, Hollywood Classics and podcasts; iii) A ranking criterion, for example, Influential, Interesting, and You Shouldn't Miss (which is equivalent to Best or Top).

Figure1 shows the block diagram of our system. The system consists of the following components:

(1) Title Classifiers, which attempts to recognize the page title of the input web page.

(2) Candidate Picker, which extracts all potential top-k lists from the page body as candidate lists.

(3) Top-K Ranker, which scores each candidate list and picks the best one;

(4) Content Processor, which post processes the extracted list to further produce attribute values, etc.

A. TOP-K LIST EXTRACTION ALGORITHM

1) Read input query from user
2) Classify the component of input query using n gram language model
3) Extract WebPages using document parsing
4) Construct document object modeling tree
5) Apply pruning technique to optimize tree
6) Classify text node using natural language processing algorithms
7) Extract feature from result using input query
8) Preprocess result
9) Generate result

Now we can consider each step of algorithm in detail:

1. Read input query from user.

2. Classify the component of input query using n gram language model
   Using n gram models of Natural Language, predict the class of the words given in the input query Classify the words of the query into noun phrase, verbs and adjectives for understanding the context of the query. Using this we can group the query syntactically as well as semantically.

3. Extract web pages using document parsing
   Extract web page source code from a particular search engine and apply string processing algorithms like pattern matching or Brute Force algorithm to parse tags and data.

4. Construct document object modelling tree:

Natural Language Processing (NLP) and information retrieval (IR) algorithms can also benefit from content extraction, as they rely on the relevance of content and the reduction of "standard word error rate" to produce accurate results. Content extraction allows the algorithms to process only the extracted content as input as opposed to cluttered data coming directly from the web. Currently, most NLP-based information retrieval applications require writing specialized extractors for each web domain. While generalized content extraction is less accurate than hand-tailored extractors, they are often sufficient and reduce labour involved in adopting information retrieval systems.

While many algorithms for content extraction already exist, few working implementations can be applied in a general manner. Our solution employs a series of techniques that address the aforementioned problems. In order to analyze a web page for content extraction, pass web pages through an open source HTML parser, open XML, which corrects the mark up and creates a Document Object Model tree. The Document Object Model is a standard for creating and manipulating in-memory representations of HTML (and XML) content. By parsing a webpage's HTML into a DOM tree, this system not only extract information from large logical units similar to Buyukkokten's Semantic Textual Units" (STUs,), but can also manipulate smaller units such as specific links within the structure of the DOM tree. In addition, DOM trees are highly editable and can be easily used to reconstruct a complete webpage. Finally, increasing support for the Document Object Model makes our solution widely portable.For this project Xerces HTML DOM is used.

5. Apply pruning technique to optimize tree
   Separate and Conquer rule always provides a simplified solution for achieving accurate results. Pruning is the common framework for avoiding the problem of over fitting noisy data. For the purpose of optimization this system is using Pruning by bound technique to remove the heterogeneous or unwanted results. This will greatly minimize the computation time of our system.

6. Top-k Extraction from the results undergoing following steps:
   a) Title Recognition
   b) List Extractor
   c) Content Extractor

IV. CONCLUSION & FUTURE SCOPE

In this paper proposed system is extracting particular information from any type of web page. This is not implemented before this. This system is having proper goal of finding top-k list. It is fulfil with knowledge and on web we get millions of this type of list available. So our algorithm can help to extract top-k list faster and provide more accurate. In future we will improve speed of extraction of top-k list.

REFERENCES

[1] Zhixian Zhang, Kenny Q. Zhu , Haixun Wang , Hongsong Li ,"Automatic Extraction of Top-k Lists from the Web",, IEEE , ICDE Conference, 2013 ,978-1-4673-4910-9.

[2] F. Fumarola, T. Weninger, R. Barber, D. Malerba, and J. Han, "Extracting general lists from web documents: A hybrid approach," in *IEA/AIE (1)*, 2011, pp. 285–294.

[3] G. Miao, J. Tatemura, W.-P. Hsiung, A. Sawires, and L. E. Moser, "Extracting data records from the web using tag path clustering," in *WWW*, 2009, pp. 981–990.

[4] M. J. Cafarella, E. Wu, A. Halevy, Y. Zhang, and D. Z. Wang, "Webtables: Exploring the power of tables on the web", in *VLDB* Auckland, New Zealand, 2008.

[5] Smith Tsang, Ben Kao, Kevin Y. Yip, Wai-Shing Ho and Sau Dan Lee, "Decision Trees for Uncertain Data",IEEE conference,2011.

[6] J. Wang, H. Wang, Z. Wang, and K. Q. Zhu, "Understanding tables on the web," in *ER*, 2012, pp. 141–155.

[7] W. Gatterbauer, P. Bohunsky, M. Herzog, B. Krüpl, and B. Pollak, "Towards domain independent information extraction from web tables", In *WWW*, pages 71–80. ACM Press, 2007.