A Framework for Deep Web Data Extraction using Vision and Novel based Approach

Namrata Bhalerao
ME (Comp.Engg) II Year Student
Department of Computer Engineering,
MGM's College of Engineering and Technology,
University of Mumbai, India.

Subhas Shinde
Associate Professor
Department of Computer Engineering,
LT College of Engineering
University of Mumbai, India.

ABSTRACT
World Wide Web has more and more online Web databases which can be searched through their Web query interfaces. The number of Web databases has reached 25 millions according to a recent survey. All the Web databases make up the deep Web (hidden Web or invisible Web). Often the retrieved information (query results) is enwrapped in Web pages in the form of data records. These special Web pages are generated dynamically and are hard to index by traditional crawlerbased search engines, such as Google and Yahoo. This kind of special Web pages deep Web pages.. Deep Web contents are accessed by queries submitted to Web databases and the returned data records are enwrapped in dynamically generated Web pages (they will be called deep Web pages in this paper). Extracting structured data from deep Web pages is a challenging problem due to the underlying intricate structures of such pages. Until now, a large number of techniques have been proposed to address this problem, but all of them have inherent limitations because they are Web-page-programming-language dependent. As the popular two-dimensional media, the contents on Web pages are always displayed regularly for users to browse. This motivates to seek a different way for deep Web data extraction to overcome the limitations of previous works by utilizing some interesting common visual features on the deep Web pages. The paper, a novel vision-based approach that is Web-page programming-language-independent is proposed. This approach primarily utilizes the visual features on the deep Web pages to implement deep Web data extraction, including data record extraction and data item extraction. We also propose a new evaluation measure revision to capture the amount of human effort needed to produce perfect extraction. Experiments on a large set of Web databases show that the proposed vision-based approach is highly effective for deep Web data extraction. The approach consists of four primary steps: Visual Block tree building, data record extraction, data item extraction, and visual wrapper generation. Visual Block tree building is to build the Visual Block tree for a given sample deep page using the VIPS algorithm. With the Visual Block tree, data record extraction and data item extraction are carried out based on our proposed visual features. Visual wrapper generation is to generate the wrappers that can improve the efficiency of both data record extraction and data item extraction. Highly accurate experimental results provide strong evidence that rich visual features on deep Web pages can be used as the basis to design highly effective data extraction algorithms.

Keywords
Deep Web /invisible web. Deep web search Engine, Web–Page programming language.

1. INTRODUCTION
Finding information about people in the World Wide Web is one of the most common activities of Internet users. Person names, however, are highly ambiguous. Finding information about people in the World Wide Web is one of the most common activities of Internet users. Person names, however, are highly ambiguous. In most cases, the results for a person name search are a mix of pages about different people sharing the same name. The user is then forced either to add terms to the query (probably losing recall and focusing on one single aspect of the person), or to browse every document in order to filter the information about the person he/she is actually looking for. In an ideal system the user would simply type a person name, and receive search results clustered according to the different people sharing that name. One particular case of this people-document association task is referred to as personal name resolution. The task is as follows: given a set of documents all of which refer to a particular person name but not necessarily a single individual (usually called referent), identify which documents are associated with each referent by that name. Different methods have been used to represent documents that mention a candidate, including snippets, text around the person name, entire documents, extracted phrases, etc.

2. RELATED WORK
There are various approaches which have been reported in related work of literature survey. Various surveys are done web related data extraction .In this section, we review on previous work on web extraction. WebOQL system [1], whose goal is to provide such a framework. The WebOQL data model supports the necessary abstractions for easily modeling record-based data, structured documents and hypertexts. Web-scale Data Integration [2], contends that traditional data integration techniques are no longer valid in the face of such heterogeneity and scale. Extracting Content Structure for Web Pages based on Visual Representation [3], new approach of extracting web content structure based on visual representation was proposed. The produced web content structure is very helpful for applications such as web adaptation, information retrieval and information extraction. Block-level Link Analysis [4]. In this paper, we proposed two novel link analysis algorithms called Block Level PageRank (BLPR) and Block Level HITS (BLHITS)

3. PROJECT ARCHITECTURE
The project work is broadly classified to 4-different modules for the development purpose as shown in Figure 1 above.
Crawler-Module: It is basically a design of a software agent that browses the World Wide Web in a methodical and

automatic manner. Based on the user's target name, the application will find and download PDF file suspicious in being lists containing the target names. The search is done with proposed technique that provides the possibility search for PDF documents only. The Crawler will get the URLs from proposed technique's reply (which contains many other fields), and download the documents to the local disc. The application will work with configurable "maximum" of documents. The default is 50 (i. e. maximum 50 PDF documents will be downloaded). The crawler performs second search for finding image results for target name. First N pictures will be taken to the indexer (N is configurable).

Indexer Module: The indexer will extract the text from the PDF document, filter documents that not seem to be relevant papers. Indexer will analyze the text and find the suitable information pieces. The first phase of the project we will focus on email, document title and document "abstract" part. As to the images, the indexer will download them to the local disc. Another problem is avoiding ambiguous names. In order to solve this problem we will need user interfere. Indexer will divide the collected information to "clusters". Cluster will be identified by information pieces type that were defined as the key. In this version the key info is the email address. Each cluster will contain the key e-mail and the rest of the information pieces found in the same documents as the key.

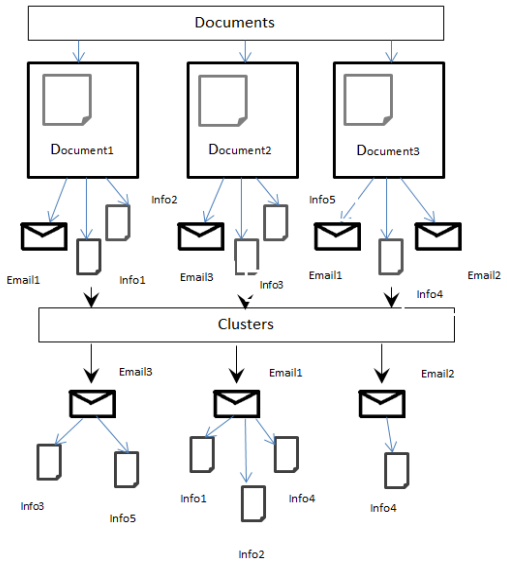

Fig 1: Project Module Description

I. Cluster Module: Information from the documents is inserted into the clusters according to the emails (keys) in the source document. 5 clusters with the biggest amount of related document are presented to the user. The user will choose the clusters and that are likely to belong to the target person. The chosen clusters are passed to the page builder. If the indexer produced only one cluster – the application will skip this step. In case that no clusters were produced – message will be shown to the user.

II. Page Builder Module: Gets the clusters and images and creates HTML page containing all the information from the clusters. The sections in the HTML page are:

   i. Header part – notification ("This page was automatically generated etc.), the target name.
   ii. Images – up to N images which were found and can be successfully shown; path to the images in the local disk.
   iii. Publications – title; abstract; URL; local path.
   iv. Contacts – emails.

When the page is ready, the default system browser is opened, and the page can be visible. The page builder also shows a summary display of the search process. The summary screen contains the number of the documents and images, the number of clusters and so on.

4. EXISTING SYSTEM AND ITS EFFECTS

Searching for information on the Web is not an easy task. Searching for personal information is sometimes even more complicated. Below are several common problems we face when trying to get personal details from the web:

- Majority of the Information is distributed between different sites.
- It is not updated.
- Multi-Referent ambiguity – two or more people with the same name.
- Multi-morphic ambiguity which is because one name may be referred to in different forms.
- In the most popular search engine Google, one can set the target name and based on the extremely limited facilities to narrow down the search, still the user has 100% feasibility of receiving irrelevant information in the output search hits. Not only this, the user has to manually see, open, and then download their respective file which is extremely time consuming. The major reason behind this is that there is no uniform format for personal information.

Maximum of the past work is based on exploiting the link structure of the pages on the web, with hypothesis that web pages belonging to the same person are more likely to be linked together.

5. PROPOSED SYSTEM

In this proposed system, we explore the visual regularity of the data records and data items on deep Web pages and propose a novel vision-based approach, Vision-based Data Extractor to extract structured results from deep Web pages automatically. Vision-based data extractor is primarily based on the visual features human users can capture on the deep Web pages while also utilizing some simple non-visual information such as data types and frequent symbols to make the solution more robust. It consists of two main components, Vision-based Data Record extractor and Vision-based Data Item extractor .By using visual features for data extraction, vision based data extractor avoids the limitations of those solutions that need to analyze complex Web page source files. In this research work, the focus is mainly on searching for personal information of scientists and researchers. The user has to set the proper target name for search, which when completed, the user will receive complete PDF and image files based on the key (e-mail) of the search. Each group of information items (cluster) will be defined by its key (email) and the user make the choice. The result page will be produced from the chosen clusters for making the search operationally accurate, we will assume the usage of IEEE doc files as they carry a standard format of name, e-mail ID, publication, images, and links to the full images.

5.1 Assumptions and dependencies

The major assumption of the project work is that the targeted PDF document file which the user wants to search over HTTP has universal same format internationally acclaimed and known to everyone. The cluster key is assumed to be email ID of the author with the help of which the user can segregate the same types of different papers published by same author. The dependency of the project work is that user needs constant connectivity of internet and JSDK software for the proper execution of the project. The user also needs to assure about enough space in their system as the end result of the project work is all the research documents in PDF format in one drive.

5.2 Functionality

The goal of data extraction framework is simple: to foster innovation in the search industry. Developers, start-ups, and large internet companies can use data extraction framework to build and launch web-scale search products that utilize the entire web search index. Data extraction framework gives you access to web's investments in crawling and indexing, ranking and relevancy algorithms, and powerful infrastructure. by combining your unique assets and ideas with our search technology assets, data extraction framework is a platform for the next generation of search innovation, serving hundreds of millions of users across the web. This project is based on web mining framework. Data extraction framework is a developer network initiative to provide an open search web services platform. The main goal and idea of data extraction framework is to give developer's free access to the web Search index. The input for the search process is the "target" name. It can be a combination of first name and last name, or the last name alone (in this case there is a greater chance for name ambiguity)

5.3 Design Considerations

The approach of the proposed system employs a four-step strategy. First, given a sample deep Web page from a Web database, obtain its visual representation and transform it into a Visual Block tree which will be introduced later; second, extract data records from the Visual Block tree; third, partition extracted data records into data items and align the data items of the same semantic together; and fourth, generate visual wrappers (a set of visual extraction rules) for the Web database based on sample deep Web pages such that both data record extraction and data item extraction for new deep Web pages that are from the same Web database can be carried out more efficiently using the visual wrappers.

6. CONCLUSION

The World Wide Web is a rapidly growing and changing information source. Due to the dynamic nature of the Web, it becomes harder to find relevant and recent information.. We present a new model and architecture of the Web Crawler using multiple HTTP connections to WWW. The multiple HTTP connection is implemented using multiple threads and asynchronous downloader module so that the overall downloading process is optimized. The user specifies the start URL from the GUI provided. It starts with a URL to visit. As the crawler visits the URL, it identifies all the hyperlinks in the web page and adds them to the list of URLs to visit, called

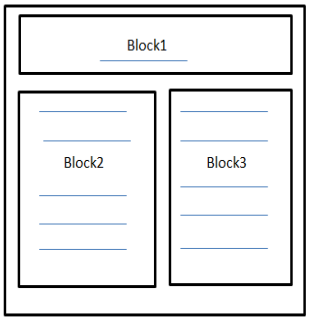

Fig 2: Structure of Deep Web Structure

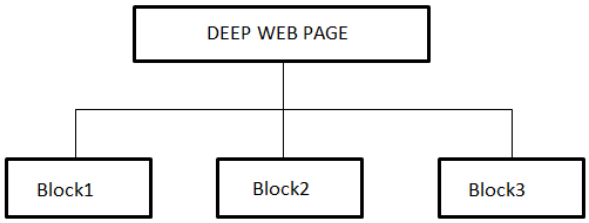

Fig 3: Visual Block Tree

the crawl frontier. URLs from the frontier are recursively visited and it stops when it reaches more than five level from every home pages of the websites visited and it is concluded that it is not necessary to go deeper than five levels from the home page to capture most of the pages actually visited by the people while trying to retrieve information from the internet. The web crawler system is designed to be deployed on a client computer, rather than on mainframe servers which require a complex management of resources, still providing the same information data to a search engine as other crawlers do.

7. FUTURE DEVELOPMENT

Web Crawler forms the back-bone of applications that facilitate Web Information Retrieval. In this paper we have presented the architecture and implementation details of our crawling system which can be deployed on the client machine to browse the web concurrently and autonomously. It combines the simplicity of asynchronous downloader and the advantage of using multiple threads. It reduces the consumption of resources as it is not implemented on the mainframe servers as other crawlers also reducing server management. The proposed architecture uses the available resources efficiently to make up the task done by high cost mainframe servers. A major open issue for future work is a detailed study of how the system could become even more distributed, retaining though quality of the content of the crawled pages. Due to dynamic nature of the Web, the average freshness or quality of the page downloaded need to be checked, the crawler can be enhanced to check this and also detect links written in JAVA scripts or VB scripts and also provision to support file formats like XML, RTF, PDF, Microsoft word and Microsoft PPT can be done.

8. REFERENCES

[1] Wei Liu and Weiyi Meng, "Vision based approach for deep web data extraction" IEEE trans. on Knowledge and Data Engineering 2010.

[2] Gustavo O. Arocena, Alberto O. Mendelzon, "WebOQL: Restructuring Documents, Databases and Webs"

[3] Jayant Madhavan, Shawn R. Jeffery, Shirley Cohen, Xin (Luna) Dong, David Ko, Cong Yu, Alon Halevy,"Web-scale Data Integration: You can only afford to Pay As You Go"

[4] Ron Bekkerman and Andrew McCallum "Disambiguating Web Appearances of People in a Social Network"

[5] Danushka Bollegala and Yutaka Matsuo," Measuring Semantic Similarity between Words Using Web Search Engines" WWW 2007 / Track: Semantic Web

[6] James Caverlee, Ling Liu, and David Buttler, "Probe, Cluster, and Discover:Focused Extraction of QA-Pagelets from the Deep Web"IEEE2004

[7] Jer Lang Hong, "Deep Web Data Extraction" IEEE2010

[8] Robert Baumgartner, Michal Ceresna and Gerald Lederm¨uller, "DeepWeb Navigation in Web Data Extraction" IEEE2005

[9] Chia-Hui Chang, Chun-Nan Hsu , Shao-Chen Lui, "Automatic information extraction from semi-structured Web pages by pattern discovery" 0167-9236/02/$ - see front matter D 2002 Elsevier Science B.V. All rights reserved. PII: S0167-9236(02)00100-8 E-mail addresses: chia@csie.ncu.edu.tw(C.H.Chang),chunnan@iis.sinica.edu.tw(C.-N.Hsu), anyway@db.csie.ncu.edu.tw (S.-C. Lui).

[10] Bing Liu, Robert Grossman, Yanhong Zhai Kai Simon, Georg Lausen, "Mining Data Records in Web Pages" liub@cs.uic.edu, grossman@uic.edu, yzhai@cs.uic.edu SIGKDD .03, August 24-27, 2003, Washington, DC, USA Copyright 2003 ACM 1-58113-737-0/03/0008.$5.00.

[11] Bing Liu, Robert Grossman, Yanhong Zhai in"Mining Data Records in Web Pages" liub@cs.uic.edu, grossman@uic.edu, yzhai@cs.uic.edu, SIGKDD .03, August 24-27, 2003, Washington, DC, USA Copyright 2003 ACM 1-58113-737-0/03/0008.$5.00.

[12] Yiyao Lu, Hai He, Hongkun Zhao, Weiyi Meng, Clement Yu in" Annotating Structured Data of the Deep Web" {ylu0, haihe, hkzhao, meng}@cs.binghamton.edu, yu@cs.uic.edu, 1-4244-0803-2/07/$20.00 ©2007 IEEE.