Journal of Network Communications and Emerging Technologies (JNCET) www.jncet.org
Volume 5, Special Issue 2, December (2015)

Surfacing and Ranking the Deep Web

Babita Ahuja
MRU, Faridabad, India.

Dr. Anuradha
YMCA University Faridabad, India.

Dimple Juneja
DIMT, Kurukshetra, India.

Abstract –Deep Web forms a major part of the WWW. Most of the existing search engines do not work exclusively to extract the data from the deep web. Day by day the websites are placing their data in the databases. So the data in the deep web is growing at a tremendous rate. The traditional search engines are neither capable of extracting the data residing in the deep web nor capable of ranking the pages from the deep web. As most of the data on WWW is present on the deep web so there is a need of technique to extract and rank pages from deep web. So we propose a new technique called the surfacing and ranking the deep web (SRDW). SRDW will extract the data from the deep web and will rank the pages extracted from the deep web.Deep Web forms a major part of the WWW. Most of the existing search engines do not work exclusively to extract the data from the deep web. Day by day the websites are placing their data in the databases. So the data in the deep web is growing at a tremendous rate. The traditional search engines are neither capable of extracting the data residing in the deep web nor capable of ranking the pages from the deep web. As most of the data on WWW is present on the deep web so there is a need of technique to extract and rank pages from deep web. So we propose a new technique called the surfacing and ranking the deep web (SRDW). SRDW will extract the data from the deep web and will rank the pages extracted from the deep web.

Index Terms – Surface Web, Deep Web, Query Interface, WWW.

This paper is presented at International Conference on Recent Trends in Computer and information Technology Research on 25th& 26th September (2015) conducted by B. S. Anangpuria Instituteof Technology & Management, Village-Alampur, Ballabgarh-Sohna Road,Faridabad.

1. INTRODUCTION

The surface web is part of the web that is easily indexed by our traditional search engines. The hidden web is not indexed by our search engines. The hidden web forms 99% of the WWW. The data residing in deep web is of high quality. The hidden web is of four different types [1]. The four types of invisibility are:

- The Opaque Web: The traditional search engines incapable of crawling such kind of web because of the depth of crawl, frequency of crawl and disconnected URLs..
- The Secret Web: This kind of web is not indexed by the traditional search engines because the website administration does wish the search engine to index them. The website administrator uses the Robot Exclusion Protocols and the Meta tag no-index to prohibit the search engines from indexing them.
- The Proprietary Web: The users are required to register first in order to access the data.
- The Proper Invisible Web: The format of files present on the deep web is a major issue. The images, pdf, audio files, video files, exe files are not indexed by the search engines. If search engines try to handle them then there are robot traps. So robot doesn't try to index them. The new pages keep on adding in the WWW. These new pages are also left without indexing. The major part of hidden web is behind the query interfaces. The search engines are not able to extract the data behind the query interfaces automatically. SRDW will surface the data behind the query interfaces web pages and will rank them.

1.1. Benefits of Hidden Web

Bright Planet [2] has quantified the size and relevancy of the deep. Their key findings include:

- The size of the deep web is 400 to 550 times the size of the surface web.

- The deep web has 7500 TB of data while the surface web has 17TB of data.
- The deep web sites receive fifty per cent greater monthly traffic than surface sites.
- Deep web content is highly relevant to every information need, market, and domain.
- The deep web is the largest growing category of new information on the Internet.

2. RELATED WORK

In the recent years, the importance of hidden web has become very renowned and considered as a very important part of WWW. The studies are conducted that are extending current-day crawlers to build repositories that include pages from the "hidden Web", the portion of the Web behind searchable HTML forms [3]. To determine the freshness of the web pages is another problem. The work has been conducted that gives a new technique to continuously update/refresh the Hidden Web repository [4]. The query-based database crawling has been modeled to fetch the data from the hidden web [5]. Jian Qiu [6] also presents new index structures for querying the hidden web. But this study again considered only single attribute. Here, clustering of data is done to compress the index. But this technique is inefficient for storing the multi-attribute based hidden web data. HiddenSeek [7] uses a keyword based indexing and searching technique for single-attribute hidden web sites. This approach uses the inverted index for indexing and searching method the hidden web data. HiddenSeek takes a term frequency of keyword as a factor for ranking the results i.e, whether the keyword appears in the URL of a page. The MetaQuerier [8] was designed by Chang, 2005. The goal of MetaQuerier is twofold. First, to make the deep Web systematically accessible, it will help users find online databases useful for their queries. Second, to make the deep Web uniformly usable, it will help users query databases. This system focuses on the query interface processing and the processing of the query results is not involved in detail.

3. PORPOSED MODELLING

In the proposed technique Surfacing and ranking deep web the user will not be required to fill thousands of query interfaces in order to get the behind them. User will fill a single search text field for his query as shown in Figure 1. The query issued by user will be processed thoroughly. The user keywords are extracted after the user query processing. User keywords are the placed in the URL templates generated on the system side as shown in Figure 2. The dynamic URL's are generated for GET method of form submission. The results behind the query interfaces are displayed to the user. For post methods of form the user keywords are embedded in the source code of the page and are submitted. The results are fetched on the fly from the website servers. These fresh results from deep web are then ranked and displayed to the user.

3.1. Modules of SRDW

The different modules of SRDW are Query Interface Extraction, URL Template Extraction of "GET" method of form submission, Template Extraction, Process the user query, Create the dynamic URL of GET method, Auto-submission of forms for POST method, and Deep Web Page Ranking.

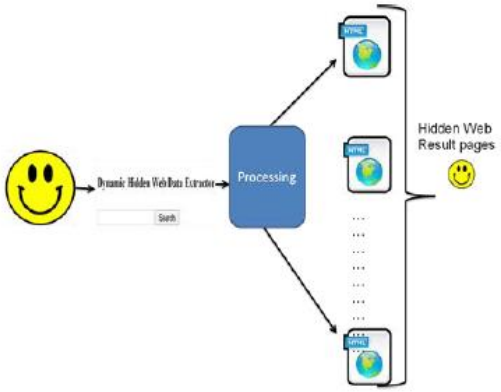

Figure 1 Steps taken by user in proposed technique SRDW

3.1.1. Query Interface Extraction

The data behind the query interfaces forms a lion's share of the data in the hidden web. The query interfaces acts as a very significant channel to access the databases residing on the server machines databases. These query interfaces are created on computer machines but the irony is that the computer machines are not able to understand them. So fetching the query interfaces is a major task. The query interfaces can have two form processing techniques GET and POST. Most of the techniques developed consider only the GET method of form submission. The proposed technique works on both the GET and the POST. In the query interface extraction module we will fetch the query interfaces of few sample domains. In order to fetch the query interfaces we will issue the query using the google API. Here we will consider two domains book and the car domains. When query is fired using google API ex: "book" then results of book domain are displayed. These result pages contain static web pages as well as pages

containing query interfaces. The pages containing the query interfaces are filtered and are stored. The stored query interfaces are further categorized on the basis of GET and POST.

3.1.2. URL Template Extraction of "GET" method of form submission.

The URL filtered above contains the query interfaces. The query interfaces having GET method of form submission will be handled in this module. When these query interfaces are submitted to servers the result pages are shown. In GET as the input field data changes so does the result page's URL changes. The URL template of these result pages will be fetched and will be stored in the database. In order to accomplish this task the form tag of the query interfaces will be extracted. A temporary web page will be created which will contain the code of form tag extracted in the previous step. The websites provides the relative path of the result pages. So these relative paths will be converted into the absolute path. The initial input field will be filled with any value. The temporary web page will be created in such a way that it will automatically be redirected to the result page. The URL of the result page will be extracted. The URL will be analyzed and a generic template of the URL will be created and stored in the URL template database.

3.1.3. Template Extraction

The query interfaces extracted in first step contains web pages having both GET and POST methods of form submission. In GET as the input field data changes so does the result page's URL changes. In GET URL template is stored. But in the POST method the result page's URL does not change on the change of the input field data. For every query interface having POST method of form submission a separate html page will be created. This html page contains only the form tag data of the original query interface. The page will be created in such a way that the input field values will be filled atomically by the value that are provided by the user while issuing the query in a single search textbox. The values will be filled automatically and the submission too will be done automatically. The fresh results from the actual websites servers will be presented to the user.

3.1.4. Process the user query

The user will issue the query in a single search text field. The user query will be processed. The tokens or keywords will be extracted from the user query. The punctuation symbols etc. will be removed from the user query. The lemmatization or stemming of user query will be done to formulate the final list of user query keywords. The keywords will be processed to identify the domain of the user query.

3.1.5. Create the dynamic URL of GET method

All the URL templates of user query domain will be picked from URL template database. The user keywords will be placed in the URL templates and the form input data fields. The URL's will be created and will be displayed to the user. When user will click on the URL then the result pages from the actual website will be fetched and will be displayed to the user after page ranking.

3.1.6. Auto- submission of forms for POST method

The web pages having form submission method will be filled automatically by the keywords of user query. After automatic form filling the web pages will be submitted automatically. The results from the deep web will be fetched and will be displayed to the user after page ranking.

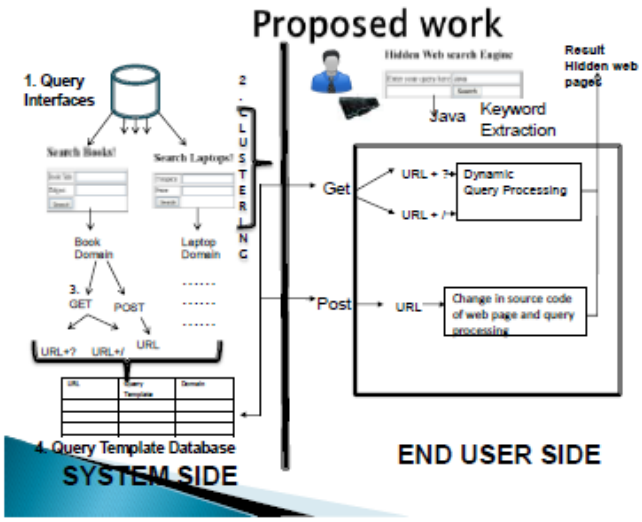

Figure 2 Proposed Architecture of Surfacing and Ranking Deep Web

3.1.7. Deep Web Page Ranking

Initially the 10 pages having maximum page rank will be displayed. For these web pages and the other web pages from the deep web the fresh page rank will be calculated. For calculating the page rank all the types of web mining will be used. Web structure Rank will be calculated by analyzing all the hyperlinks in the pages. Web Content rank will be calculated by analyzing all the images, contents of the web pages. Web usage rank will be calculated by accessing the web server logs. After the proper scrutiny of the web logs the Web usage rank will be calculated. So the final rank of the

web page will be the sum of web structure rank, web content rank and web usage rank. The pages will be ordered on the basis of the rank calculated and will be displayed to the user.

4. RESULTS AND DISCUSSIONS

The proposed technique SRDW has many advantages over the exiting techniques. The advantages of SRDW are given below:

1. Million of query interfaces need not to be filled

2. Mass storages of data is not required as results are fetched on the fly

3. No indexing is needed

4. No frequent crawling is needed as results are fetched from the website servers dynamically

5. Fresh results are displayed to the user. The results are not saved locally on the servers. Always the fresh results are displayed to the users.

6. Proposed system works for both the GET and POST method of form submission.

5. CONCLUSION

The proposed technique SRDW uncovers the data hidden behind the query interfaces which are having both GET and POST method of form processing. The pages from deep web are ranked and are displayed to the user.

REFERENCES

[1] Chris Sherman and Gary Price Hidden Web. "Uncovering Information Sources Search Engines Can't See". CyberAge book November 2001.
[2] Bergman, Michael K. White Paper. "The Deep Web: Surfacing Hidden Value". Journal of Electronic Publishing Volume 7, Issue 1, August, 2001.
[3] Sriram Raghavan Hector Garcia-Molina, "Crawling the HiddenWeb" Computer Science Department Stanford University Stanford,USA.
[4] Rosy Madaan "A Framework for Incremental Hidden Web Crawler" (IJCSE) International Journal on Computer Science and Engineering Vol. 02, No. 03, 2010, pp. 753-758.
[5] Ping Wu Ji-Rong Wen, Huan Liu, Wei-Ying Ma "Query Selection Techniques for Efficient Crawling of Structured Web Sources"
[6] Jian Qiu, Feng Shao, Misha Zatsman, Jayavel Index Structures for Querying the Deep Web, Workshop on the Web and Databases (WebDB), 2003, pp. 79-86
[7] Ntoulas, A., Zerfos, P., Cho, J. Downloading Textual Hidden Web Content through Keyword Queries, In Proceedings of the 5th ACM/IEEE Joint Conference on Digital Libraries, 2005.
[8] Chang, K; He, B; Zhang, Toward Large Scale Integration: Building a MetaQuerier over Databases on the Web, 2005.