www.ijecs.in
International Journal Of Engineering And Computer Science ISSN: 2319-7242
Volume 5 Issues 6 June 2016, Page No. 16889-16890

Ontology based indexing of hidden web: Review

Manvi Siwach, Sushmita Singh
Department of Computer Engineerin YMCA University of Science & Technology Faridabad, India
sushmi278@gmail.com

Department of Computer Engineering YMCA University of Science & Technology
Faridabad, India manvi.siwach@gmail.com

Abstract
We think is that what we can access easily on the internet is whole of the information available. But actually the amount of information available is much larger in volume. Because the web is divided into two parts surface web which we can access easily and the deep web which is invisible to us as the search engines are unable to index it directly. To handle this huge volume of information, Web searcher uses search engines. But hidden web contains a large collection of data that is unreachable by normal hyperlink-based search engines. To access this content, one must submit valid input values to the HTML form. The resultant web pages from the previous step are stored and then to improvise the search results for a user query we need indexing techniques. Indexing of hidden web is done to reveal the relevance of the document according to the context of the search. The objective is to find out different ways for indexing the web pages from the hidden web using ontology and analyze the different techniques that are previously proposed. The main advantage of storing an index is to optimize the speed and performance while finding relevant documents from the search engine storage area for a user given search query.

Keywords: *Hidden web, Deep web, Indexing, Ontology, Attribute extraction.*

Introduction

Recently the World Wide Web (WWW) has grown into a vast shelter for multi-domain data and information. As a consequence, there is an explosion in information. This information thus needs to be handled very carefully. The World Wide Web is divided into two parts: surface web (or visible web) and deep web (or the hidden web). Deep web covers almost 90% of the whole web. It is the part of web whose contents are not indexed by the standard search engines. The routine crawlers rely on the hyperlinks on the Web to discover pages, thus current search engines cannot index the Hidden-Web pages. To access this content, different techniques are used to submit valid input values to the HTML form and then extract the content hidden behind the HTML form. Web researchers have introduced various types of Web-page indexing mechanism to retrieve web-pages from Web-page repository. Ontology is a formal representation which includes vocabulary for referring to the terms in that particular domain and logical statements that describe the relationships among the terms. Automatic attribute extraction focuses on discovering the importance of web page according to the context based on the keywords. This can be further utilized for the purpose of indexing (these extracted attributes will be used to map the query with the related URL).

The main objective of this paper is to highlight and analyze different techniques for attribute extraction and indexing the hidden web data sources. Section 2 contains the Literature work, Section 3 contains the comparison between different proposed techniques, Section 4 contains the Conclusion and future work and finally the last section contains the references.

2. Literature Work

Attribute may be defined as a basic element of the web page that describes the domain of the web page. Extracting that attribute can prove to be an important step in indexing of the deep web. However several indexing techniques involve this step and some do not. Below are few attribute extraction techniques-

Automatic Attribute Extraction from deep web data sources (Yo Jung An, James Geller, Yi Ta Wu, Soon Ae Chun)

- Extracting attributes from the deep web data source both from the programmer's point of view and user's point of view that is PVA and UVA respectively. In order to automatically extract the attributes for each Web data source, we have used a three-stage algorithm.
- Given a set of Web data sources, the PVAs (what the programmer can see in the code of the webpage) are obtained from the inner identifiers of all the Web data sources. Secondly, the UVAs (what the user can see on the web page) are obtained from the free text within the query interface.
- Also the semantic redundancy (different words with same meaning, e.g grown-up and old means same) is resolved by ontology (WordNet). Finally, the final attributes (FAs) of each Web data source are determined based on PVAs and UVAs.
- The algorithm consists of four main sub-algorithms
  PVA extraction
  UVA extraction
  Ontology based attribute expansion
  Final attribute extraction

Limitations:

- There is a need to add more semantics to the processing.
- Domain specific ontology would have been more beneficial.

An improved Extraction Algorithm from Domain Specific Hidden web (Juhi Sharma, Mukesh Rawat)

- .The aim of this paper is to automate the process of accessing the hidden data. Here firstly the html web pages are fetched and stored in a repository and then these html web pages are analyzed for html forms, search interfaces and other components.
- The basic structure has four repositories - HTML page repository, Form repository, Search interface repository, Hidden web repository for storing the respective components of the webpage and the final result in the hidden web data repository.
- There are five modules in the algorithm
  - Fetching the pages,
  - Analyzing the form
  - Extracting the search interface
  - Fetching the labels
  - Submitting the form.
- There is a label-value set which further helps in extracting the hidden data by automatically filling the html form.

Limitations:

- Ontology is not used which may lead to several issues.
- Also the module for search interface extractor simply relies on the label of the buttons on the page which may give poor results.

2.1. Indexing Techniques

1. **Web page Indexing based on the Prioritized Ontology Terms((Sukanta Sinha, Rana Dattagupta, Debajyoti Mukherjee, TCS, Kolkata, India)**

   - This paper proposes a new concept of prioritizing the ontology terms. For this term relevance value is calculated for each term on the basis of number of occurrences, weight value, term synonyms existing in the web page.
   - The word with highest relevance value is phrased as dominating

ontology term. And the successive maximum relevance values are sub-dominating ontology terms.

- Each word, say w has a primary attachment which contains the page-ids of pages that have w as dominating ontology term and a secondary attachment that contains the page ids of the pages that have w as sub-dominating ontology term. Webpage ids are added to the attachments till all the web pages are indexed.
- The time complexity of this algorithm is O(log k) where 'k' is number of ontology terms.

Features:

- It reduces the search string parsing time as it directly selects the search tokens.
- Highly scalable.
- The number of indexes is lesser than a general search engine which saves time and index storage cost.

2. Context based indexing in Search Engines using Ontology(Parul Gupta, Dr. A.K Sharma, YMCAUST, Faridabad)

- Basic search engines follow three steps- Gathering web documents (Crawler), Parsing through the hyperlinks across the web (Spider), Creating a highly efficient index (Indexer).
- The authors are presenting an algorithm for improvising the indexing part of the process i.e the third step. Simple term based indexing faces polysemy (means a word has multiple meanings) and synonymy (means that multiple words having the same meaning).
- Here first the context of the page is extracted based on the keyword. Thesaurus and context repository are used to extract the context of the webpage. But multiple contexts are extracted from a single keyword from a webpage then using ontology repository and the relationships the particular context related to the web page is extracted.
- Finally an index is created with three columns one with context second with related terms and third with the document ids that contain those terms.

Features:

- This algorithm gives 60% better results than normal term based indexing.
- Basing the technique on context rather than keyword helps in improving the quality of the retrieved results.
- Also helps in disambiguating the meanings of homonyms.

3. Conception and use of Ontologies for Indexing and Searching by Semantic Contents of Video Courses (Merzougui Ghalia, Djoudi Mahieddine, Behaz Amel)

- This paper deals with indexing of video documents which is much more complicated than text documents as it is not easy to

decompose it into identifiable units.

- For this purpose two ontologies are constructed one is for the domain of teaching and the other is the pedagogical ontology of a video course. First ontology defines relation between different concepts like teaching, addressing, depending etc. Second ontology defines relation between slide, video course, pedagogical object, etc.
- The annotating process describes each element as a concept in the field starting with localization and temporal segmentation using the OntoCoV tool. Conceptual index is presented which poses queries on the temporal segments of video.
- As concepts are used instead of words the weight of concept is calculated CF_ISDF (Concept Frequency_Inverse Segment and Document Frequency).Then comes the searching phase which is the final step and gives the result in order of relevance.

Feature:

- If a concept appears in one or two segments at most it will have high weight and value.
- The system returns a sorted list of segments with corresponding lesson, beginning, duration and the pedagogical objects included in the segment. Thus the user can select the required segment.
- Though the approach is yet to evolve still it explains the feasibility and importance of ontology in pedagogical video courses.

4. **Ontology based text indexing and querying for the semantic web (Jacob Kohler, Stephen Philippi, Michael Specht, Alexander Ruegg)**

- The authors are aiming to develop a search engine that uses several mapped RDF ontologies for concept based indexing. The paper basically shows how HTML based internet is linked to RDF based semantic web by linking words in text to concepts of ontologies.
- First the ontologies are imported using RDF-parser then the equivalent concepts of different ontologies are mapped. Secondly spiders search and store web pages and then the normal keyword indexing is applied to the web pages.
- Next step links the words in the index formed by the previous step to ontological concepts. This step supports word sense disambiguation (mouse as a pointing device vs. mouse as an animal).
- The results of the different indexing processes are stored in a relational database. ER diagrams are used for representing the data structure. The frontend serves users to use the index in order to search web pages using keywords and ontological concepts.

Features:

- This allows the seamless integration of domain specific ontologies for concept based information retrieval from different domains.
- This method helps in differentiating between homonyms and thus the user receives same result when he

uses different conjugation forms of the same word.

3. Comparison table

Sr. no.	Paper	Problem with homonyms	Attribute extraction used	Techniques used	Relevance	Text document or video document
1	Webpage Indexing based on the Prioritized Ontology Terms	not solved	yes	relevance value calculation, domain specific search, ontology	relevant	text
2	Context based indexing in Search Engines using Ontology	Solved	only based on frequency	ontology, thesaurus, context extraction	more relevant as context is also the matter of concern	text
3	Conception and use of Ontologies for Indexing and Searching by Semantic Contents of Video Courses	not solved	based on temporal segments of video	pedagogical ontology, ontology, segmentation, conceptual indexing	relevant although yet to be evolved	video
4	Ontology based text indexing and querying for the semantic web	Solved	no	ontology mapping, ontology indexing, supervised and unsupervised disambiguation, spidering	Highly relevant	text

4. Conclusion

This paper represents different techniques of indexing the hidden web with the corresponding techniques and algorithms .There are several methodologies with different complexities and different techniques but a better technique is needed with lesser complexity and higher relevancy to the hidden web. The methods that are not evolved yet have the provision if implementation and some methods have the provision of extendability in different domains.

References

- http://en.wikipedia.org/wiki/
- AKSHR: A Novel Framework for a Domain-specific Hidden Web Crawler(Komal Kumar Bhatia, A.K. Sharma, Rosy Madaan Department of Computer Engineering, YMCA Institute of Engineering, Faridabad)
- Design of an Ontology Based Adaptive Crawler for Hidden Web (Manvi.M, K.K.Bhatia, A.Dixit)
- Hidden Web Data Extraction using Wordnet Ontology's(VidyaSagar Ponnam, V. P Krishna Anne, Venkata Kishore Konki)

- Automatic Attribute Extraction from deep web data sources(Yoo Jung An, James Geller, Yi Ta Wu, Soon Ae Chun)
- Webpage Indexing based on Prioritized Ontology Terms(Sukanta Sinha, Rana Dattagupta, Debajyoti Mukherjee, TCS, Kolkata, India)
- Conception and use of Ontologies for indexing and searching by Semantic content of video courses(Merzougui Ghalia, Djoudi Mahieddine, Behaz Amel, Batna University, Algeria)
- An Improved Extraction Algorithm from domain specific Hidden web(Juhi Sharma MIET Meerut, Mukesh Rawat MIET Meerut)
- Ontology based text indexing and querying for the semantic web(Jacob kohler, Stephan Philippe, Michael Specht, Alexander Ruegg)
- Context based Indexing in Search Engines using Ontology (Parul Gupta, Dr. A.K.Sharma)