Query Intensive Interface Information Extraction Protocol for Deep Web

Dilip Kumar Sharma
Reader- Dept. of Computer Science & Engineering
G L A Institute of Technology and Management,
Mathura, India
todilipsharma@rediffmail.com

A. K. Sharma
Professor & Head-Dept. of Computer Engineering
YMCA Institute of Engineering,
Faridabad, India
ashokkale2@rediffmail.com

***Abstract*— A new Query Intensive Interface Information Extraction Protocol (QIIIEP) for deep web retrieval process is proposed. Auto query word extraction and auto form unification procedure are newly proposed in order to comprehend various functions of the proposed protocol. Proposed protocol offers great advantages in deep web crawling without over burdening the requesting server. However, conventional deep web crawling procedures result in heavy communication processing loads and procedural complexity for applying either schema matching or improper otology based query. This makes it difficult to crawl entire contents of deep web. In the proposed protocol, the trade-off between correct query response and communication loads is solved by generating knowledge base at QIIIEP server. Therefore, the proposed protocol can realize flexible and highly efficient data extraction mechanism after deploying QIIIEP server on deep web domain. It enables not only the one stop information retrieval process but also provides auto authentication mechanism for supplied domain.**

Keywords- Deep web, HTTP, QIIIEP, query interface analysis, values allotment, response analysis and navigation, relevance ranking.

I. INTRODUCTION

A large part of the Web is "hidden" behind search forms and is indexed only by typing a set of keywords, or queries, to the forms. These pages are known as the Hidden Web or the Deep Web as search engines generally cannot index the deep web pages and do not show them in the results. Searching the deep web is difficult process because each source searched has a unique method of access. Hidden web crawlers must also provide input in the form of search queries. This raises the issue of how best to equip crawlers with the necessary input values for use in constructing search queries. To overcome these issues an open framework based protocol for deep web retrieval process is proposed for simultaneous searches. It supports the current trends in the field of deep web information retrieval process which consist of four steps i.e. Query interface analysis, values allotment, response analysis & navigation and relevance ranking. Proposed protocol will reduce complexity in these activities except relevance ranking of deep web data querying.

II. RELATED WORKS

Pages for search interfaces are commonly HTML forms which is filled and submitted by users and server respond accordingly. But every form is not search interface. Search form can be identified by using one of the simplest method i.e. heuristic rules [1], [2]. Other approaches to detect search interface are decision trees based classification models to detect search interface and random forest algorithm where classification is made by aggregating predictions of individual decision trees in the forest in which each classifier is realized from a subset of the feature space. The aggregated approach can fully exploit the useful features in search forms [3].

The QIIIEP (Query Intensive Interface Information Extraction Protocol) will eliminate this step by self guiding to the crawler about the search interface.

After detection of hidden web search interface, the next task is to identify accurate matching for finding semantic correspondences between elements of two schemas. Many automatic or semi-automatic matching systems meticulous in a simple 1:1 matching, such as Cupid method [4], OMA method[5], GLUE and LSD method[6][7] and Similarity Flooding method[8], for schema extraction are considered for the non-hierarchical structure of query interface, which neglects the grouping and hierarchical relationships of attributes. So the semantics of a query interface cannot be captured correctly. Based on the nonhierarchical model, literatures [9], [10] proposed a hierarchical model and schema extraction approach which can group the attributes and improve the performance of schema extraction of query interface. But they show the poor clustering capability of pre-clustering algorithm due to the simple grouping patterns and schema extraction algorithm and require human interaction and not suitable for dynamic large scale data sets. Other approaches are DCM [11] and MGS framework [12] which pursues a correlation mining approach by exploiting the co-occurrence patterns of attributes, and proposes a new correlation measure while other hypothesizes that every application field has a hidden generative model and can be viewed as instances generated from models with possible behaviors [13].

978-1-4244-4711-4/09/$25.00 ©2009 IEEE

New schema extraction algorithm Extr[14] which is based on the pre-clustering of attributes P by using MPreCluster, Komal Kumar Bhatia et al [15] presented in his research literature that mapping can be done by using domain specific interface .

The QIIIEP (Query Intensive Interface Information Extraction Protocol) will reduce this complexity by using pre-information about the form and its elements from QIIIEP server. The knowledge base is generated by auto query word extractor or it is provided by site administrator.

Ontology is a formal specification of a shared conceptualization [16]. This step is required for analyzing area or specialization of web page so that in further steps appropriate data set will be efficiently placed in query part of the page. Deitel et al. [17] present an approach for learning ontology from RDF annotations of Web resources. Stojanovic [18] presents an approach for an automated migration of data-intensive web sites into the semantic web. The paper [19] presents an approach TANGO (Table Analysis for Generating Ontologies) to generating ontologies based on HTML table analysis. Zhiming Cui et al published his research [20] which makes Mini-Ontology from Query Interfaces by applying employs vision-based approach.

Otology identification is not required in proposed protocol because QIIIEP server's pre knowledge about the form and its elements will be enough for choosing correct value for each element of form.

Integration of the databases with the query interfaces is further step in this process. The search form interface brings the attributes together and this step will analyze appropriate data values by their structural characteristics of the interface and the order of attributes in the area as possible as it can. For integrating interfaces, the core part is dynamic query translator, which can translate the users' query into different forms [21] [22]. Mapping is done by Fuzzy comprehensive evaluation methods [23] which map the attribute of the form to the data values.

Query Translation Technique is used to get query from different deep web sources i.e. to translate queries to sources without primary knowledge. Some methods can be concerned such as type-based search-driven translation framework by leveraging the "regularities" across the implicit data types of query constraints. In [24] they found that query constraints of different concepts often share similar patterns, and encoded more generic translation knowledge for each data type. Type-based predicate mapping method [25] proposed by Z.Zhang focusing on text type attribute with some constraint.

The QIIIEP (Query Intensive Interface Information Extraction Protocol) will reduce this complexity by using QIIIEP server's query words database which is generated by auto query word extractor or it is provided by site administrator.

III. PROPOSED WORK

The QIIIEP (Query Intensive Interface Information Extraction Protocol) is an application-level protocol for semantic otology based query word composition, identification and retrieval systems. It is based on request/response semantics. This specification defines the protocol referred to as QIIIEP 1.0. It will work on port 55555 on http server and generate response encoded by using XML.

The initial specification of this protocol can be discussed on the basis of figure1 given below.

1. In first step, crawler will request for any web server to fetch a page.
2. In second step, crawler will analyses the form to identify search interface. Search interface must include rel tag to describe the QIIIEP server address and form id.
3. In this step, crawler will send the request to QIIIEP server for getting the semantic query word list which is defined by the site administrator or QIIIEP auto query word extractor to correlate the form fields.
4. In this step, QIIIEP server will reply to the crawler about each entry of that form.
5. In this step, crawler will send the filled form by placing received query words to the HTTP server.
6. In this step, crawler will crawl the contents generated by that query word.

a. In this step, QIIIEP auto query word extraction module continuously watch the form interface to extract query word supplied by user as well as from the content generated by processed query.

b. Finally, it will store the query word into the QIIIEP database for further analysis.

c. It merges the form ids to the forms at the time of form generation.

d. Fetch the form id to query word table relationship from QIIIEP server.

A. *HTTP Server*

The http server is regular http server on which the web site is deployed.

B. *Auto form id generation module*

This module will help to implement QIIIEP protocol on current architecture of web site. It will parse every form of that web site and merge the form id with QIIIEP server query word list so that at the time of crawler request the crawler will properly identify the search interface for sending the request of keywords to the QIIIEP server.

C. *Auto query word extraction module*

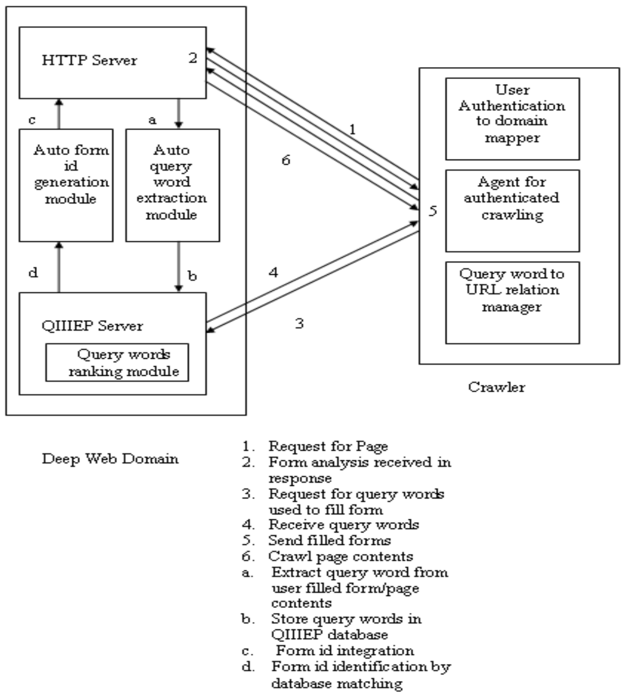
This module will extract the query words supplied by users of that site so that QIIIEP server can extend the query word list by exploiting advantage of human curiosity to find relevant information on that domain.

D. Query word ranking module

This module is responsible for providing the best match query word assignment to the form filling process and reduce the over loading by less relevant querying to the domain.

E. User Authentication to domain mapper

This module of crawler is responsible for mapping the login credentials provided by users to the crawler with the information provider domain. The main benefit of using this mapper is to overcome the hindrance of information retrieval in between result link and information. The crawler will use the mapping information to allow the specific person to receive information contents directly from the domain by automatic login procedure and reduce a step of separate login for user.


HTTP Server
2
c
a
Auto form id generation module
Auto query word extraction module
d
b
QIIIEP Server
Query words ranking module
1
6
5
4
3
User Authentication to domain mapper
Agent for authenticated crawling
Query word to URL relation manager
Crawler
Deep Web Domain

1. Request for Page
2. Form analysis received in response
3. Request for query words used to fill form
4. Receive query words
5. Send filled forms
6. Crawl page contents

a. Extract query word from user filled form/page contents
b. Store query words in QIIIEP database
c. Form id integration
d. Form id identification by database matching

Figure 1 Basic architecture of QIIIEP

F. Agent for authenticated crawling

This module work for those sites where the information hidden by the authentication form. It stores authentications credentials of every domain in its knowledge base situated at crawler, which is provided by the domain administrator. At the time of crawling it will automatically authenticate it self on the domain for crawling the hidden contents. These contents will be used only to indexing keywords and will not be available by search service as catch because of privacy issue.

G. Query word to URL relation manager

This module will store each and every query word associated with specific element of form by creating reference of domain path, so that at the time of link generation in response to search, query word can be mapped to provide the contents by sending query word in post request to the domain.

IV. IMPLEMENTATION ISSUES

This protocol is still in it's initial specification stage but proposed framework's mechanism is not only very simple to implement but also imposed lesser amendments in existing infrastructure because all the functionality is implemented over http server so there is no need to resolve any major issue before experimenting the propose framework. The crawler will send the request for keywords in xml encoded document which is having information of url and form-id QIIIEP sever will also respond this request in xml encoded document which is transferred by secure network. Further it will use guidelines inherited from OAI -PMH for achieving request- response XML document formulation. The QIIIEP server can be deployed on to the same server or it can be deployed on separate server.

V. COMPARISON WITH OTHER PROTOCOLS

Some of other supporting mechanisms are also proposed to deep web retrieval processes in which frameworks are designed for extraction of information. Search/Retrieval via URL (SRU) protocol is a standard XML-focused search protocol for Internet search queries that uses Contextual Query Language (CQL) for representing queries. The SRU uses the REST protocol and introduces sophisticated technique for querying databases by simply submitting URL-based queries.
For example
URL?version=1.1&operation=retrieve&query=dilip&maxRecords=15

Proposed protocol is deployed on the querying server as well as on crawler so there is no need to make request from get method only .The crawler will find the contents by using post method of HTTP protocol.

The Open Archives Initiative (OAI) [26] Protocol for Metadata Harvesting (OAI-PMH) provides an interoperability framework based on the harvesting or retrieval of metadata from any number of widely distributed databases. Through the services of the OAI-PMH, the disparate databases are linked by a centralized index. The data provider agrees to have metadata harvested by the service provider. The metadata is then indexed by the harvesting service provider and linked via pointers to the actual data at the data provider address.

Proposed protocol is simple to implement and controlled by site administrator so the main advantage is that the contents provided to crawler is controlled by authorized entity. A separate repository for meta data is not required in this implementation.

VI. RESULTS

We implemented initial version of this protocol specification on www.qiiiep.org and analyzed the results. We used three different test domains to judge the precision of retrieved contents.

TABLE I

Domain	Query Forms	Correct identification
Auto	3	3
Book	4	4
Job	6	6

Form identification statistic

TABLE II

Domain	Form ids	Query words received	Successful contents extraction
Auto	37723d89ce0b05185c3e052af8e4b6d6	38	36
	55c64ad2f0d6a6ef4388b33b54123868	12	10
	868f92d7b327a7163c73d771067b6cd4	16	16
Book	7c563ab917356cc029db94fcfdce34b5	13	13
	fc4597a57d0cfcef9aa82162cab8e080	15	12
	e0bfffd062a28403d9662610be421e0d	44	34
	8b555f930bf01516abc059ba5f9c384e	17	12
Job	61fab2718c890a61bcf87772dc66dfa2	5	4
	c1c1841bfd2c4554235183bc0de2e836	55	42
	de54cd34b8a2ebcd2b852f410331df4a	15	13
	67c27add7110dbe020bb401a4672565e	11	11
	200d21f16a9b95df5eaa746a02a67f1d	16	16
	aa502d15a3135934a964640ddd7abec0	34	32

Query words and contents extraction statistic

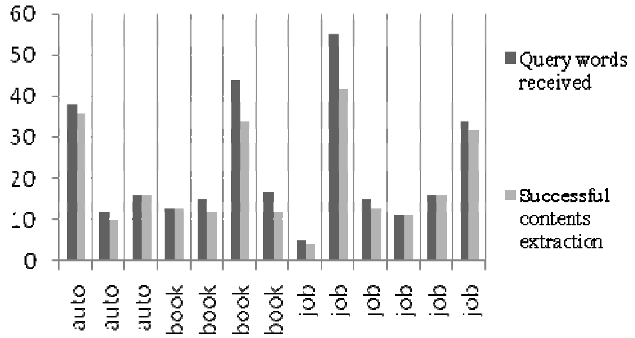


Figure 2 Comparison of success with no. of query words at different domain

As shown above the graph is plotted in between received query words and successful content extraction at different domain, and concluding that contents extraction are close to query words received at a satisfactory level for all three domains.

VII. CONCLUSIONS

A theoretically justified open framework based Query Intensive Interface Information Extraction Protocol is proposed in this paper, which may eliminate deficiencies such as improper mapping, over traffic load from unmatched semantic query etc. for deep web information retrieval process. Most of the time the best search engines for a site are the ones which are written by those who knows the contents the best [27]. Future work includes the design and implementation analysis of this theoretically proposed open framework protocol by analyzing every aspect of its architectural and implementation specifications, considering the fact that it must allow simple implementation with minimum modification in existing ongoing web architecture.

ACKNOWLEDGMENT

All the companies/products/services names are used for identification purposes and may be the trademarks of their respective owners.

REFERENCES

[1] S.Raghavan, H. Garcia-Molina, "Crawling the hidden web". In: Proceedings of the 27th International Conference on Very Large Data Bases, Roma, Italy, 2001.

[2] J. P. Lage, A.S. Dasilva, P.B. Golgher & A. H. F. Laender, "Automatic generation of agents for collecting hidden web pages for data extraction," Data & Knowledge Engineering, vol. 49, pp. 177–196. 2004.

[3] X. B. Deng, Y.M. Ye , H.B. Li, J. Z. Huang, "An Improved Random Forest Approach For Detection of Hidden Web Search Interfaces," In: Proceedings of the Seventh International Conference on Machine Learning and Cybernetics, IEEE, Kunming, 2008.

[4] J. Madhavan, P.A. Bernstein, E. Rahm, " Generic Schema Matching with Cupid" In: 27th VLBB Conference,Rome, 2001.

[5] H.H. Do, E. Rahm, " COMA-a System for Flexible Combination of Schema Matching Approaches," In: Proc.28th Intl. Conference on Very Large Databases (VLDB), Hong Kong August 2002.

[6] A.H. Doan, P. Domingos, A. Levy, “Learning source descriptions for data integration,” In: Proceeding WebDB Workshop, pp. 81-92. 2000.

[7] A.H. Doan, P. Domingos, A. Halevy, “Reconciling schemas of disparate data sources: a machine-learning approach,” In: Proc ACM SIGMOD Conference, pp. 509-520. 2001.

[8] S. Melnik, H. Garcia-Molina, E. Rahm, “Similarity Flooding: A Versatile Graph Matching Algorithm,” In: Proceeding l8th International Conference on Data Engineering (ICDE), San Jose Feb. 2002.

[9] W. Wu, A. Doan, C. Yu, “Integrating Deep Web data sources,” Dissertation

[10] W. Wu, C. Yu, A. Doan, W. Meng, “An interactive clustering-based approach to integrating source query interfaces on the Deep Web,” In Proceedings of the ACM SIGMOD International Conference on Management of Data (SIGMOD’04), pp. 95–106. 2004.

[11] B. He, K. C.-C. Chang, J. Han, “Automatic complex schema matching across web query interfaces: A correlation mining approach,” In: Technical Report UIUCDCS-R-2003-2388, Dept. of Computer Science, UIUC, Dec. 2003.

[12] B. He, K. C.-C. Chang, “Statistical schema matching across web query interfaces” In: SIGMOD Conference, 2003.

[13] X. Zhong, Y. Fu, Q. Liu, X. Lin, Z. Cui, “A Holistic Approach on Deep Web Schema Matching,” In: International Conference on Convergence Information Technology, IEEE, 2007.

[14] B. Qiang, , J. Xi, B. Qiang, L. Zhang, “An Effective Schema Extraction Algorithm on the Deep Web,” IEEE, 2008.

[15] K. K. Bhatia, A.K. Sharma, “A Framework for Domain Specific Interface Mapper (DSIM),” In: IJCN International Journal of Computer Science and Network Security Vol. 8. No. 12, Dec 2008.

[16] M. Niepert, C. Buckner, C. Allen, “A Dynamic Ontology for a Dynamic Reference Work,” JCDL’07, Vancouver, British Columbia, Canada, 2007.

[17] A. Deitel, C. Faron, R. Dieng, “Learning ontologies from rdf annotations,” In: Proceedings of the IJCAI Workshop in Ontology Learning, 2001.

[18] L. Stojanovic, N. Stojanovic, R. Volz, “Migrating data intensive web sites into the semantic web,” In: Proceedings of the 17th ACM symposium on applied computing, pp. 1100– 1107. 2002.

[19] Y. A. Tijerino, D. W. Embley, D. W. Lonsdale, Y. Ding, G. Nagy, “Towards ontology generation from tables,” World Wide Web Journal, pp. 261–285. 2005.

[20] Z. Cui, P. Zhao, W. Fang, C. Lin, “From Wrapping to Knowledge: Domain Ontology Learning from Deep Web,” In: International Symposiums on Information Processing, IEEE, 2008.

[21] X. Meng, S. Yin, Z. Xiao, “ A Framework of Web Data Integrated LBS Middleware,” Wuhan University Journal of Natural Sciences, 11(5), pp. 1187-1191. Nov. 2006.

[22] H. He, W. Meng, C. T. Yu, Z. Wu, “WISE-Integrator: An Automatic Integrator of Web search interfaces for e-commerce,” In Proceedings of the 29th International Conference on Very Large Data Bases (VLDB’03), pp. 357–368. 2003.

[23] Z. Zhang, B. He, K.C.C. Chang, “Light-weight Domain-based Form Assistant: Querying Web Databases On the Fly,” VLDB Conference, Trondheim, pp. 97-108. Norway 2005.

[24] S. Chen, L. Wen, J. Hu, S. Li, “Fuzzy Synthetic Evaluation on Form Mapping in Deep Web Integration,” In: International Conference on Computer Science and Software Engineering, IEEE, 2008.

[25] B. He, Z. Zhang, , K.C.C. Chang, “Meta Querier: Querying Structured Web Sources On the-fly,” In: Proceedings of SIGMOD, System Demonstration, Baltimore, Maryland June 2005.

[26] The Open Archives Initiative Protocol for Metadata Harvesting (Protocol Version 2.0), http://www.openarchives.org/OAI/2.0/openarchivesprotocol.htm, 2003.

[27] From the OpenSearch FAQ, http://opensearch.a9.com/docs/faq.jsp