ISSN Online: 2319 – 9253
Print: 2319 – 9245

EFFICIENT INFORMATION EXTRACTION USING WEB SERVICES

#1T.Manuja- M.Tech Student,#2 G.Vijay Kumar – Assistant Professor,

Department of Computer Science and Engineering,

G.Pulla Reddy Engineering College, Kurnool, A.P.

Abstract: The API of the web service restricts the query answering that a service can answer. Consider a web service that might provide a method that returns the different book publishers of a given book but it might not provide the different books of a given publisher even though the backend database contains the required information. This is an asymmetric Problem. To solve the problem of asymmetry, Information Extraction (IE) can be done by collecting the correct input values as parameter bindings for the web service. NER technique is used to find the required entities in text documents. Inverse functions should be prioritized to get the information for low budget of function calls. The main aim is to improve the performance and compute maximum results and reduce budget by minimizing the rewritings of initial query.

Key terms : Information Extraction(IE), NER technique, Web Service.

I.INTRODUCTION

The Application Programming Interface provides interaction between the software components and also Graphical User Interface components. The API is used to write the code and interface with other code by third parties such as Google maps API, Twitter API etc. The application Programming interface contains a collection of protocols, routines and data structures helps to build applications. The Web service API provides restriction to answer for different types of queries .The data is restricted to provide security to the database and prevent the unwanted users to access the entire data in the database.

Web services are defined by W3C used to provide interaction between machines over a network. A Web service uses UDDI and WSDL. A web service method contains the code to return the books of a given author, but does not return the authors of a given book. If the user asks for the author of a specific book, the web service cannot be called even though the database contains information. This is asymmetric Problem. Information Extraction can be done by collecting the values as parameter bindings for the web service. There are many web services for online book shopping (www.flipkart.com , www.amazon.com), about movies (www.erosnow.com), about music (www.indiamp3.com), about online shopping (www.jaboong.com) and about games (www.flipline.com). Consider a web site www.indianmp3.com offers a Web service to access its database.

II.RELATED WORK

The approach of [1] is limited to the single input argument of the function is unbound. If multiple inputs arguments are used for function the results becomes harder to estimate. The methodology of “Answering queries using templates with binding patterns”[3] rewrites first query into a set of queries to be executed over given views. When sources are incomplete, it is needed to find maximal contained rewritings of first query to provide maximum number of answers. The Magic Set Algorithm [16] decreases the number of sub-queries in a bottom – up evaluation. ANGIE system [12] answers queries using views and also prioritizes the function calls. Google’s surfacing technique [23] is used to actualize Deep Web by finding input values for forms. SUSIE uses the Named Entity Recognition Technique [29-31] to detect needed entities in text documents. Information extraction is the extraction of structured data from documents. Extracted data is too noisy to allow direct querying. Information Extraction methods are Wrapper Induction [32], Entity Extraction [31, 36] or Fact extraction [33-35].

Existing System:

Existing system exists a conjunctive query plan over the views. This plan is equivalent to the original query is NP-hard in the query's size. This plan assumes that views are complete. But the sources may overlap or complement with each other. when sources are incomplete , composing the existing functions to compute answers requires high budget before any answer is returned supposing to find maximum contained rewritings of the first query and to provide maximum number of answers.

Proposed System:

Information Extraction is done to collect values that can be used as parameter bindings for the web service. The minimum rewritings have been proposed to decrease the number of accesses by providing correct input values for asymmetric web services and to use Information Extraction to predict bindings for the input variables and validate the bindings by the web service.

Functions:

1. **Query Answering**
2. **Information Extraction**
3. **Web services**
4. **Extracting Candidates**

Query Answering:

Query answering using views without unlimited access patterns [3] is problematic . The approach of [3] rewrites the first query into a group of queries for execution over specific views. A conjunctive query and a group of views are used in the general schema. Keyword Searching is used to answer a query.

Information Extraction:

Information extraction (IE) is extraction of Structured data from Unstructured and Semi structured documents. IE is used to find needed entities and used as inputs for Web service function calls. Named Entity Recognition (NER) technique[29–31] is used to predict required entities in structured documents and used to develop candidates for the system. This effective technique searches for noun phrases among different entity names that are stored in a knowledge base.

Web Services:

Information extraction is done by guessing and validating argument bindings for the input variables passed into the web service method. Inverse functions should be prioritized.

Benefits of System:

- The current system improves the performance by using NER technique to derive needed candidates.
- The budget of calls can be reduced by reducing the rewriting of the query and maximizing the results. .

Preliminaries:

Consider the database tables such as details of author, book and publisher .Consider the variables a, i ,t, pi, p denoting the author, book id, title of the book, publisher id, publisher name respectively. Variable b denotes the binding value in that position and f denotes that there is no restriction.

The function *getBooks* can be written as

$$\mathbf{\mathit{getBooks^{bfff}\ (a,i,t,pi) \leftarrow hasTitle(\ i,t), author(a,i),publish(pi,i,p)}}$$

The function *getPublisher* can be written as

$$\mathbf{\mathit{getPublisher^{ff}(a,p) \leftarrow author(a,i),\ publish(pi,i,p)}}$$

author

author	bookId
Kurana	1
Monna	2
Joham	3

Table 1.Author details

book

bookId	title	issn
1	C	123
2	C++	134
3	Java	400

Table 2.Book details

Onpublish

pubId	pubName	bookId
1	Mc-Gill	1
2	GK	2
3	UG	3

Table 3.Publisher details

Incomplete functions:

Web services are mostly incomplete.

getPublisherb f (pi,i,p) ← author(a,i), publisher(pi,i,p), nb(a)

where nb(a) is a global relation that holds authors of books.

Database functions:

Database functions are incomplete since there are only f-adorments but no

b- adornments.

Queries: A query is a datalog rule as

Q(Y)←r1(Y1), r2(Y2),….rn(Yn)

Where r1,r2,..rn are database relations and Xi are tuples of variables or constants.

To find the author of book

q(a)←hasTitle(i, Java), author(a,i)

The answer to the query is a=Joham.

III.EXECUTION PLANS

Goal: Functional calls are used to answer queries. To compute maximum number of calls using the specific budget of calls.

Execution Plans:

Consider the following query to ask for books by Joham:

q$_1$(p)←author(joham,i), publish(pi,i,p)

we cannot access the author and publication tables directly.

Add Inverse Functions:

Let us consider the query for author and publication of book Java:

q$_2$(p,a) ← hasTitle(i,Java), onPublish(i,p), author(a,i)

we have following functions :

- ***getAuthorInfo$^{b\,f\,f\,f}$ (i,t,a,p) ← hasTitle(i,t), author(a,i), onPublish(i,p)***
- ***getRelBooks$^{b\,f\,f}$ ($t_1,i_2\,,t_2$) ← influenced(i_1,i_2), hasTitle(i_1,t_1), hasTitle(i_2,t_2)***

Let us add following function :

getAuthor$^{b\,f}$ (t,a) ← author(a,i),hasTitle(i,t)

This function retrieves the author of a book.This has same structure as *getBooks*$^{b\,f\,f}$ but a different binding pattern. We call it as inverse function of *getBooks*$^{b\,f\,f}$.

Query Answering:

Inverse Rules: The datalog program proposed in [11] is used for the function.

getRelBooks$^{b\,f\,f}$ (t_1,i_2,t_2) ←influenced(i_1,i_2), hasTitle(i_1,t_1), hasTitle(i_2,t_2)

To construct inverse rule, the function body consists of a dom atom for all bound variables of function.

influenced(f_1,i_2) ← dom(t1),
getRelBooks$^{b\,f\,f}$(t_1,i_2,t_2)
hasTitle(i_2,t_2) ←dom(t2),
getRelBooks$^{b\,f\,f}$ (t_1,i_2,t_2)
hasTitle(f_1,t_1) ← dom(t_1),
getRelBooks$^{b\,f\,f}$(t_1,i_2,t_2)
where f1= f(t_1,getRelBooks) replace variable i_1.

Before function calling, all the input parameters of a function call are bound .

SMART FUNCTION CALLS:

A smart function call is a function call whose inputs come from a query or previous function calls and whose outputs are subset of outputs of previous functional calls.

getAuthor$^{b\,f}$ *(Java,a)* is a smart function call since its ouputs *({author(a,Java)})* are subset of outputs of following calls.

String Searching Algorithms:

String searching algorithms are used to find the occurrences of a pattern in a text. Different algorithms are used for the String searching. KMP algorithm has time complexity is O(m+n). KMP algorithm, in the worst case analysis, the number of comparisons is $O(n + rm)$, where r is the total number of matches. Boyer-Moore hits wrost case in a binary alphabet, so KMP is safer. The KMP algorithm is used to search for same pattern in different text documents.The main demerit of the Boyer-Moore algorithms is the preprocessing time and the space requirement based on the alphabet size and the pattern size. If a given pattern is small i.e1 to 3 characters length, then it is a good choice to use the Naive algorithm. Consider the alphabet size is large, the Knuth-Morris-Pratt algorithm is a good choice. In all the other situations, for large texts, the Boyer-Moore algorithm is best to use. At last, the Boyer-Moore-Horspool is the best algorithm, with respect to execution time, for all pattern lengths. The running time of shift-or algorithm is similar to the KMP algorithm. The merit of this algorithm is search for general patterns such as "don't care" symbols, complement of a character, etc. uses same searching time but the preprocessing time is different.

IV.ALGORITHM

String Matching Algorithm

The main goal of String searching or String Matching is to find the location of a specific text pattern in a text .

The Knuth-Morris-Pratt Algorithm

- The Knuth-Morris-Pratt (KMP) String Matching Algorithm collects the information from previous comparisons.
- A failure function (*f*) is computed to estimate the usage of the previous comparison if it fails.

Pseudo code:

- Algorithm KMPMatch(E,P)
- **Input :** String E (text) with ‘n’ number of characters and P (pattern) with ‘m’ number of characters.
- **Output :** Starting index of the first substring of E matches P, otherwise the pattern didn’t match with substring of E.
- f ← KMPFailureFunction(P) {builds the failure function}
- a←0
- while a< n do
- if P[b] = E[a] then
- if b = m - 1 then
- return a - m - 1 { a match is found}
- a←a + 1
- b ←b + 1
- else if b> 0 then {no match found}
- b← f(b-1) { b indexes after prefix matches in P}
- else
- a← a+ 1
- return “no substring of E that matches P ”

Time Complexity Analysis

- Denote $k = a - b$
- In every iteration throughout the while loop, anyone of three conditions will be satisfied.
- If $E[a] = P[b]$, then k remains the same, a increases by 1.
- If $E[a]$!= $P[b]$ and $b = 0$, then *a and k both* increases by 1 ,since b remains the same.
- If $E[a]$!= $P[b]$ and $b > 0$, then a remain same and k increases minimum by 1, since k changes from $a – b$ to $a - f(b -1)$.
- For every iteration in the loop, either a or k increases by at least 1, so that maximum number of loops is $2n$.
- KMPFailureFunction is $O(m)$ Preprocessing time and $O(n)$ to search for pattern in string.
- Total Time Complexity: $O(n + m)$

Boyer-Moore Horspool (BMH) Algorithm:

This algorithm uses only the Bad character shift rule. The best case time complexity is O(n).and Boyer-Moore algorithm uses both Good Suffix rule and Bad Character Shift rule.

Regular Expressions:

Regular expressions are used to search for a particular pattern in a text. For example phone number , card number, Gmail account name, website name etc. Regular expressions uses concatenations, alterations and repititions.

Named Entity Recognition (NER):

Named Entity Recognition technique is used to find the required entities in text documents and categorize the elements based on categories such as person names, location, organization, time and quantities.

Natural Language Processing (NLP):

Natural Language Processing is related to the interaction between machines and human. NLP includes the tasks such as Named Entity Recognition, query answering, Speech recognition, subjective analysis, parsing a given sentence , Information Extraction etc.

LDAP:

LDAP is a Light Weight Directory Access Protocol used to check the information from a server. LDAP is an Internet protocol used for email service.

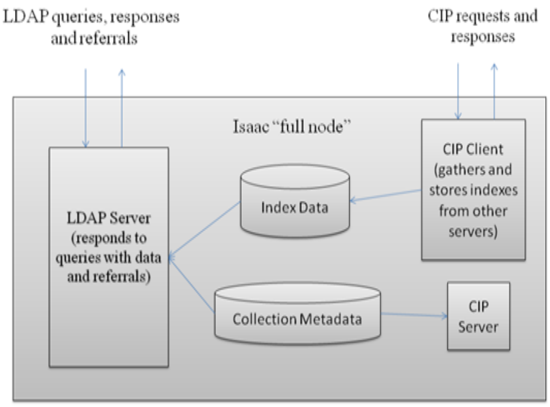
LDAP queries, responses and referrals
CIP requests and responses
Isaac “full node”
LDAP Server (responds to queries with data and referrals)
Index Data
CIP Client (gathers and stores indexes from other servers)
Collection Metadata
CIP Server

Figure 1.Query Processing

CIP (Client IP Address): CIP is a small API that allows different applications on a server to retrieve a client's local IP Address. The API contains a COM interface used in several languages such as Visual Basic, C++, .NET etc.

V.FUTURE WORK

The current approach can answer the queries if a single input is passed as an argument in a function. For multiple inputs to pass through the function, string matching algorithms should be used for future work. Boyer-Moore algorithm is used for searching multiple patterns in a text.

VI.CONCLUSION

This system has been developed as a solution for problem of asymmetry web services. Considering the real time web services information extraction is done to determine bindings for the input variables and validate these bindings by the Web service. Prioritization of Inverse functions. Information Extraction and Web services are used naive information extraction algorithms. Future work can be done for searching new algorithms and the discovery of new Web services ,WCF services and their integration into the system.

REFERENCES

[1]F. M. Suchanek, G. Kasneci, and G. Weikum, "YAGO: A Core of Semantic Knowledge," in WWW, 2007.
[2] S. Auer, C. Bizer, G. Kobilarov, J. Lehmann, R. Cyganiak, and Z. Ives,"DBpedia: A nucleus for a Web of Open Data," Semantic Web, 2008.
[3] A. Rajaraman, Y. Sagiv, and J. D. Ullman, "Answering queries using templates with binding patterns," in PODS, 1995.
[4] C. Li and E. Y. Chang, "Query planning with limited source capabilities,"in ICDE, 2000.
[5] C. Li, "Computing complete answers to queries in the presence of limited access patterns," VLDB J., 2003.
[6] S. Kambhampati, E. Lambrecht, U. Nambiar, Z. Nie, and G. Senthil,"Optimizing recursive information gathering plans in EMERAC," J.Intell. Inf. Syst., 2004.
[7] A. Cal`ı and D. Martinenghi, "Querying data under access limitations,"in ICDE, 2008.
[8] A. Deutsch, B. Lud¨ascher, and A. Nash, "Rewriting queries using views with access patterns under integrity constraints," Theor. Comput. Sci.,2007.
[9] A. Nash and B. Lud¨ascher, "Processing unions of conjunctive queries with negation under limited access patterns," in EDBT, 2004.
[10] A. Cal`ı, D. Calvanese, and D. Martinenghi, "Dynamic query optimization under access limitations and dependencies," J. UCS, 2009.
[11] O. M. Duschka, M. R. Genesereth, and A. Y. Levy, "Recursive query plans for data integration," J. Log. Program., 2000.
[12] N. Preda, G. Kasneci, F. M. Suchanek, T. Neumann, W. Yuan, andG. Weikum, "Active Knowledge : Dynamically Enriching RDF Knowledge Bases by Web Services. (ANGIE)," in SIGMOD, 2010.
[13] R. Fagin, L. M. Haas, M. A. Hern´andez, R. J. Miller, L. Popa, and Y. Velegrakis, "Clio: Schema mapping creation and data exchange," inConceptual Modeling: Foundations and Applications, 2009.
[14] C. T. Kwok and D. S. Weld, "Planning to gather information," in AAAI/IAAI, Vol. 1, 1996.
[15] S. Abiteboul, R. Hull, and V. Vianu, Foundations of Databases.Addison-Wesley, 1995.
[16] F. Bancilhon, D. Maier, Y. Sagiv, and J. D. Ullman, "Magic sets and other strange ways to implement logic programs," in PODS, 1986.
[18] W. Gatterbauer, P. Bohunsky, M. Herzog, B. Kr¨upl, and B. Pollak,"Towards domain-independent IE from web tables," in WWW, 2007.
[19] H. Elmeleegy, J. Madhavan, and A. Y. Halevy, "Harvesting relationaltables from lists on the web," PVLDB, 2009.
[20] M. J. Cafarella, A. Y. Halevy, Y. Zhang, D. Z. Wang, and E. Wu,"Uncovering the relational web," in WebDB, 2008.
[21] M. Benedikt, G. Gottlob, and P. Senellart, "Determining relevance of accesses at runtime," in PODS, 2011.
[22] M. Benedikt, P. Bourhis, and C. Ley, "Querying schemas with access restrictions," PVLDB, 2012.
[23] J. Madhavan, D. Ko, L. Kot, V. Ganapathy, A. Rasmussen, and A. Y.Halevy, "Google's deep web crawl," PVLDB, pp. 1241–1252, 2008.
[24] W. Wu, A. Doan, and C. T. Yu, "Webiq: Learning from the web to match deep-web query interfaces," in ICDE, 2006.
[25] X. Jin, N. Zhang, and G. Das, "Attribute domain discovery for hidden web databases," in SIGMOD, 2011.
[26] L. Barbosa, H. Nguyen, T. H. Nguyen, R. Pinnamaneni, and J. Freire,"Creating and exploring web form repositories," in SIGMOD, 2010.
[27] W. Liu, X. Meng, and W. Meng, "Vide: A vision-based approach fordeep web data extraction," IEEE Trans. Knowl. Data Eng., pp. 447–460,
2010.
[28] M. A. Hearst, "Automatic acquisition of hyponyms from large textcorpora," in ICCL. Association for Computational Linguistics, 1992.
[29] J. Zhu, Z. Nie, J.-R. Wen, B. Zhang, and W.-Y. Ma, "2d conditional random fields for web information extraction," in ICML. ACM, 2005.
[30] J. D. Lafferty, A. McCallum, and F. C. N. Pereira, "Conditional random fields: Probabilistic models for segmenting and labeling sequence data,"in ICML. Morgan Kaufmann Publishers Inc., 2001.
[31] H. Cunningham and D. Scott, "Software architecture for language engineering," Nat. Lang. Eng., 2004.

[32] N. Kushmerick, "Wrapper induction for information extraction," Ph.D.dissertation, U. Washington, 1997.

[33] E. Agichtein, L. Gravano, J. Pavel, V. Sokolova, and A. Voskoboynik,"Snowball: a prototype system for extracting relations from large textcollections," SIGMOD Records, 2001.

[34] M. Banko, M. J. Cafarella, S. Soderland, M. Broadhead, and O. Etzioni,"Open Information Extraction from the Web," in IJCAI, 2007.

[35] F. M. Suchanek, M. Sozio, and G. Weikum, "SOFIE: A Self-Organizing Framework for Information Extraction," in WWW, 2009.

[36]Nicoleta Preda, Fabian Suchanek, Wenjun Yuan, Gerhard Weikum "SUSIE: Search Using Services
and Information Extraction",IEEE transactions,2013.

[37]www.orion.lcg.ufrj.br/Dr.Dobbs/books/book5/chap10.html

AUTHORS PROFILE

T. Manuja, pursuing M.Tech , Dept of CSE, From G. Pulla Reddy Engineering College (Autonomous), Kurnool,AP. She completed M.Sc.IT in Vivekanadha College of Engineering and Technology for Women, Tiruchengode, Salem, Tamilnadu. She is interested in research areas Data mining, Image Processing, Data warehousing, Network Security, Natural Language Processing.

G. Vijay Kumar, Assistant Professor, Department of CSE, G. Pulla Reddy Engineering College (Autonomous), Kurnool, A.P. He is **pursuing Ph.D** in Computer Science & Engineering from Jawaharlal Nehru Technological University, Anantapur, A.P. in India. He completed B.Tech and M.Tech Degree in Computer Science & Engineering from Jawaharlal Nehru Technological University, Hyderbad, AP, in India, in 2002, and 2006 respectively. His research interests include Mobile ad hoc networks, cross-layer design,Network security and Wireless mesh networks.