Domainwise Extraction of Information and Knowledge using Automated Query Generation

Akash R. Dumbre, Yogesh S. Sandbhor, Hrishikesh V. Pardeshi, Nikhil K. Niwdunge,

Computer Department, Government College of Engineering and Research Awasari (kd), Pune

***Abstract*- Information extraction systems which are implemented traditionally have focus on extracting particular type of information. Most of them are suitable for static databases. In these approaches process of extraction has to be applied on entire database which results in slow processing of user queries. Also, efficiency and quality of extraction of specific results is poor due to huge reprocessing time. In this new approach we are going to use database queries for information extraction to minimize reprocessing time of data. Also it performs automated query generation using natural language processing. To improve performance of extraction data will be stored domainwise and will be extracted domainwise, for which user's logs will be maintained. Thus to provide efficiency & quality of extraction results incremental approach is used. It also supports multiple languages and heterogeneous databases.**

***Keywords*- Automatedquery generation, Incremental approach, Domain selection, Stemming, Conflation.**

I. INTRODUCTION

It is estimated that each year more than 600,000 articles are published in the biomedical literature, with about 19 million publication entries being stored in the Medline database. To extract information from such a large corpus of documents, it is necessary to automate query generation..For fetching concise, structured information from natural language text,methods are needed. Examples of such structured information are the extraction of entities and relationships between entities. IE is typically seen as a one time process for the extraction of a particular kind of relationships of interest from a document collection. Information extraction usually refers to identification of instances of particular events and relationships in unstructured natural language text documents. The extracted structured records can be used to populate a relational table for answering queries and running data mining tasks. Information Extraction refers to the automatic extraction of structured information such as entities, relationship between entities from unstructured sources.

A. Existing System

Traditional information extraction is usually developed as a pipeline of special purpose programs which involves sentence splitters, tokenizers, stopword removal. Such information extraction approaches are often based on file system of which main purpose is to utilize large amount of processed data between components. But it is proved that such frameworks are suitable for one time extraction that means they are not suitable for repeatedly done extraction. Example includes processing of dynamic web pages. If we use existing extraction framework we have to reprocess the entire text collection which result in expensive computation. So ideal framework should perform domainwise extraction using incremental extraction approach to reduce preprocessing time.

B.Proposed System

We propose new framework for information extraction that uses database management system as main component of this framework. This framework serves for dynamic extraction needs over file based extraction systems.For text processing named entity recognizers and stopward removal is deployed for entire text corpus. Then intermediate processed data will be stored in relational database using user logs .This avoids the need of reprocessing the entire collection of data. In this system intermediate processed data will be stored & using simple SQL insert statements new knowledge and information can be extracted. Here each sentence is processed separately& then other components are processed which results in reduced processing time. For this incremental extraction approach is used.

Traditionally implemented Information Extraction systems process entire text corpus even when small part of collection of text is affected or changed. This causes large reprocessing time which in turn reduces efficiency and quality of extraction results. To extract structured text from unstructured text domainwise extraction is necessary. Also to minimize or reduce preprocessing time of extraction incremental approach should be used.

II. TOOLS USED IN IMPLEMENTATION

Java Server Pages (JSP) technology enables you to mix regular, static HTML with dynamically generated content. You simply write the regular HTML in the normal manner, using familiar Web-page-building tools. You then enclose the code for the dynamic parts in special tags, most of which start with <% and end with %>.Different type of database software is used to record a database, Such as oracle, My SQL, MS-Access, Sybase etc. Using this database software creates a different type of database.

III. SYSTEM DESIGN

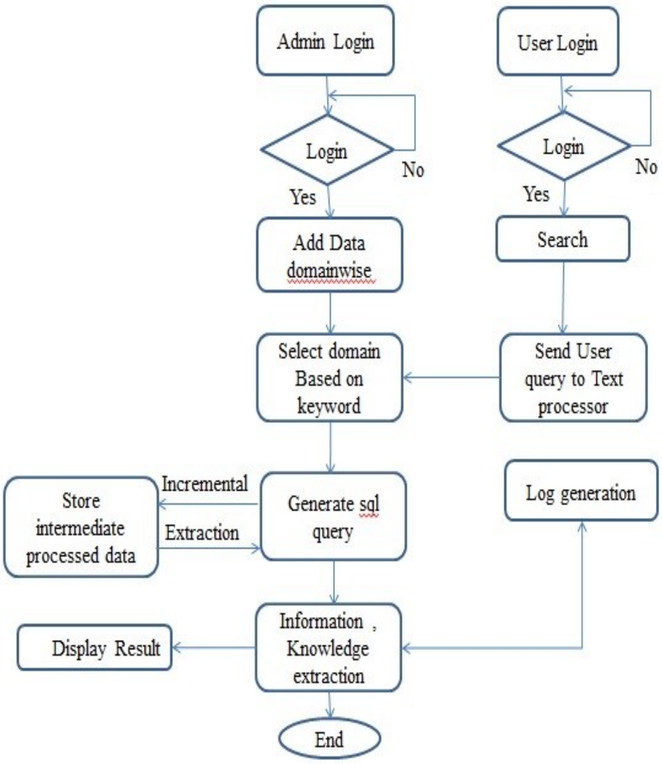

Fig.1.System Design

In our proposed system approach of incremental extraction is used. Information extraction frameworks which are traditionally implemented are considered only for static corpora. In our system, whenever extraction goal emerges, extraction has to be applied only on affected

Area of text document rather than on entire text corpus. Here information is stored in database domainwise to reduce preprocessing time and increase efficiency.

Here admin can add data to database domainwise. When user fires query in English that query is processed by text preprocessor. At this position stemming algorithm is applied. Based on the output of text preprocessor sql query is generated. When sql query is generated it is fired on the database to select promising domain from which required data is extracted.

IV. MODULES DESCRIPTION

1. *Client side Application*: Using AWT / Swing. This GUI shall allow the user to log in and search the information using natural language.
2. *Domain Selection*: A module that will allow server to select data from required specific domain to minimize preprocessing time and increase efficiency of extraction.
3. *Database management–* A database containing log information of all users is maintained on server.
4. *Client Server Interface:* A module to allow the client application to call Server.

V. ADVANTAGES OF SYSTEM

1. In this extraction, system generates log for each user which is used for incremental extraction.
2. If queries are written manually, it will be time consuming and labour intensive process which in turn results in unsatisfactory extraction. To avoid this automated query generation is used.
3. The use of incremental extraction reduces the preprocessing time .

VI. EXPERIMENTAL EVALUATION

A new paradigm for information extraction proposed here, In this extraction framework, intermediate output of each text processing component is stored so that only the improved component has to be deployed to the entire corpus. Extraction is then performed on both the previously processed data from the unchanged components as well as the updated data generated by the improved component. Performing such kindof incremental extraction can result in a tremendous reduction of processing time. To realize this new information extraction framework, we propose to choose database management systems over file-based storage systems to address the dynamic extraction needs.

Our system implements text processor which takes care of the processing of user query. Text processor removes stopwords from the user query by using conflation algorithm. After removing the stopwords user query is translated SQL query. Admin is allowed to store data into the database. He also has rights to delete the data. Based on keywords from user query appropriate domain is selected to extract information user needs. For each user log will be generated to implement incremental extraction.

The proposed key phrase extraction method consists of following Algorithms:

A. Stemming Algorithm

The application of conflation techniques to single-word terms is a way of considering the different lexical variants as equivalent units for retrieval purposes. One of the most widely used non-linguistic techniques is that of stemming algorithms, through which the inflectional and derivational variants are reduced to one canonical form. Stemming or suffix stripping uses a list of frequent suffixes to conflate words to their stem or base form. Two well-known stemming algorithms for English are the Lovins (1968) and the Porter (1980).

In addition to these approaches, it is possible to group multi-word terms within a context, assigning specific indicators of relationship geared to connect different identifiers, so that noun phrases (NPs) can be built (Salton and McGill, 1983). NPs are made up of two or moreconsecutive units, and the relationships between or among these units are interpreted and codified as endocentric constructions, or modifier-head-structures.

When conflation algorithms are applied to multi-word terms, the different variants are grouped according to two general approaches: term co-occurrence and matching syntactic patterns. The systems that use co-occurrence techniques make term associations through different coefficients of similarity. The systems that match syntactic patterns carry out a surface linguistic analysis of certain segments or textual fragments.

In the process of stemming variant forms of a word are reduced to a common form.

For example,

Connections
Connectionless →Connect
Connected.

These variant forms of words can be reduced to a common word connect using stemming algorithm. Variable part is called 'endings' or 'suffix'. Taking off these endings is called as stemming and residual part is called as stem. Endings fall into two classes:

Grammatical- The addition of -s in English to make a plural is an example of a grammatical ending. The word remains of the same type. There is usually only one dictionary entry for a word with all its various grammatical endings.

Morphological-Morphological endings create new types of word. In English -ise or -ize makes verbs from nouns (`demon', `demonise'), -ly makes adverbs from adjectives (`foolish', `foolishly'), and so on. Usually there are separate dictionary endings for these creations.

VII. CONCLUSION

In existing extraction framework it is necessary to reprocess entire text collection, which is computationaly expensive. When certain components in the pipeline or extraction goals are changed we used stored intermediate processed data. Which contains parse tree and semantic information. With the use of parse tree our framework performs extraction on the text corpus which is to be in the natural sentences such as biomedical literature . The incremental extraction approach saves more time compared to performing extraction by first processing each sentence one at a time with linguistic in our novel framework parsers and then other components. To further reduce a user's effort to perform information extraction we design two algorithms to automatically generate extraction queriesin the presence and in the absence of training data, respectively our framework have overhead of storage of the intermediate process data.

VIII. FUTURE WORK

Future work can be continued by developing better algorithms than stemmer and conflation algorithms to increase efficiency and quality of information extraction so that total preprocessing time can be reduced.

IX. ACKNOWLEDGMENT

The satisfaction that accompanies that the successful completion of any task wouldbe incomplete without the mention of people whose ceaseless cooperation made it possible, whose constant guidance and encouragement crown all efforts with success. We aregrateful to our project guide Mrs.Nemade Mam for the guidance, inspiration and constructive suggestion that helpful us in the preparation of this project. I wish to extend mysincere gratitude to Mr. D.J. Periera, HOD, Department of Computer Engineering for his valuable guidance and encouragement which has been absolutely helpful in successfulcompletion of this project work.

References

[1] F. Peng and A. Mccallum, Accurate information extraction from research papers using conditional randomelds, in In HLTNAACL, 2004, pp. 329336.

[2] H. Cunningham, D. Maynard, K. Bontcheva, and V.Tablan, GATE: A Framework and Graphical Development Environment for Robust NLP Tools and Applications, Proc. 40th Ann. Meeting of the ACL, 2002.

[3] E. Agichtein and L. Gravano, Snowball: Extracting Relations from Large Plain-Text Collections, Proc. Fifth ACM Conf. Digital Libraries, pp. 85-94, 2000.

[4] E. Agichtein and L. Gravano, Querying Text Databases for Efficient Information Extraction, Proc. Intl Conf. Data Eng. (ICDE), pp. 113-124, 2003..

[5] S. Bird, Y. Chen, S.B. Davidson, H. Lee, and Y. Zhen, Extending XPath to Support Linguistic Queries, Proc. Workshop Programming Language Technologies for XML(PLAN-X), 2005.

[6] H. Cunningham, D. Maynard, K. Bontcheva, and V.Tablan, GATE: A Framework and Graphical Development Environment for Robust NLP Tools and Applications, Proc. 40th Ann. Meeting of the ACL, 2002.

[7] M. Cafarella, D. Downey, S. Soderland, and O. Etzioni,Knowitnow: Fast, Scalable Information Extraction fromthe Web, Proc. Conf. Human Language Technology and Empirical Methods in Natural Language Processing (HLT 05), pp. 563-570, 2005.

[8] R.Krishnamurthy, Y. Li, S.Raghavan, F.Reiss, S.Vaithynathan and H.Zhu, "SystemT: A System for Declarative Information Extraction," ACM SIGMOD Record, vol.37, no.4, pp.7-13,2009.

[9] A. Doan, J.F. Naughton, R. Ramakrishnan, A. Baid, X. Chai, F. Chen, T. Chen, E. Chu, P. DeRose, B. Gao, C. Gokhale, J. Huang, W. Shen, and B.-Q. Vuong,"Information Extraction Challenges in Managing Unstructured Data," ACM SIGMOD Record, vol. 37, no. 4, pp. 14-20, 2008

[10] Luis Tari, PhanHuyTu, JorgHakenberg, Yi Chen, Tran Caoson and ChittaBaral,"Incremental Information Extraction Using Relational Databases," Proc IEEE Transaction on Knowledge and Data Engineering, Vol 24th,pp.86-99,2012.

[11] F. Peng and A. Mccallum, Accurate information extraction from research papers using conditional randomelds, in In HLTNAACL, 2004, pp. 329336.

[12] S.Sarawagi, "Information Extraction," Foundations and Trends in Databases, vol.1. no.3, pp.261-377, 2008

[13] F.Suchanek, G. Ifrim, and G. Weikum, "LEILA: Learning to Extract Information by Linguistic Analysis," Proc. ACL Workshop Ontology Learning and Population, pp. 18-25,2006.

Authors:

Akash Rajesh Dumbre Pursuing B.E. (Computer Engineering) University of Pune Department of Computer Engineering, Government College of Engineering and Research Avasari (Khurd), Taluka- Ambegaon, Dist- Pune

Yogesh Shantaram Sandbhor Pursuing B.E. (Computer Engineering) University of Pune Department of Computer Engineering, Government College of Engineering and Research Avasari (Khurd), Taluka- Ambegaon, Dist- Pune

HrishikeshVasant Pardeshi Pursuing B.E. (Computer Engineering) University of Pune Department of Computer Engineering, Government College of Engineering and Research Avasari (Khurd), Taluka- Ambegaon, Dist- Pune

Nikhil Kondibhau Niwdunge Pursuing B.E. (Computer Engineering) University of Pune Department of Computer Engineering, Government College of Engineering and Research Avasari (Khurd), Taluka- Ambegaon, Dist- Pune