Study Of Multidomain Query Optimization And Answering

Heena Agrawal
Student, M.Tech, Computer Science and Engineering,
Ramdeobaba College of Engineering and Management,
Nagpur, India
heena.agrawal12@yahoo.com

A.M.Karandikar
Assistant Professor,Computer Science and Engineering,
Ramdeobaba College of Engineering and Management,
Nagpur,India
aartikarandikar@gmail.com

Abstract— In queries having multiple domains it is seen that general purpose search engines are not able to answer multidomain queries and one of the domain is considered by specific search services but no integrated framework is obtainable. Queries which are answerable by combining knowledge from two or more domains are multidomain queries. This paper presents an overall view for multidomain queries on the web. It integrates different kind of services like web service, web search and combination of them. It introduces the well defined model for expressing query results in ranking order. Here the query tends to find the results with cross domain joins to address the solution for optimized answering of multidomain queries. This paper provides the survey for multidomain query answering with answers as an optimized solution to obtain better search results with using web services. Here the output is obtained by chunking the whole query in several domain and then combining the whole result.

Keywords- Multidomain query, query optimization relevance, reranking, semantic score, web search, web service.

I. INTRODUCTION

Multi-domain queries are the queries which can be answered by combining knowledge from two or more domains. The recent evolution of the Web is characterized by an increasing number of search engines and query interfaces, ranging from generic ones (Google) to domain-specific ones (geo-localization services or on-line catalogs). An increasing amount of search services available on web work in isolation; their intrinsic limit is the inability to support complex queries ranging over multiple domains.

If we consider If we consider a query involving multiple domains, such as “find all database conferences held within six months in locations whose seasonal average temperature is 28 ^{0}C and for which a cheap travel solution exists” , requires combining search engines specialized over different domains, for instance: (i) finding interesting conferences in the desired timeframe on online services made available by the given scientific community; (ii) finding if the conference location is served by low-cost flights; (iii) finding if there are luxury and cheap hotels in proximity of the conference location [1].

Web services are the method of sharing data and functionality among loosely-coupled systems [2]. This paper gives the research terms that comes in developing and optimizing a query system for multiple-domain web queries stressing on specific features in them. Web search is the general search like google search and also web service is the specific service based search whose goal is to create a framework in which applications distributed across the internet can interoperate through a set of standard protocols like web links or url’s, etc. Here we deal with search services where the answers are in ranking order. Page rank is the link analysis that that assigns a numerical weighting to each element of a hyperlinked set of documents.

In this way , in this paper we means to develope a multidomain query answering where queries are expressed as execution strategies over web services by doing the scheduling of service invocations. Results are returned in ranking order with reference to the semantic score of number of occurrences of the query keywords into the web based data for that search. In rest of the paper we gives the survey on the work done on the optimized multidomain query answering.

The remaining paper is formed as follows: Section II gives literature review of different terms involved i n multidomain query optimized answering.

II.Literature Review

A. System Architecture

The framework given in [1] is for query answering. The architecture is defined in fig.1. The user had a query on the global ontology, equipped with a set of mappings with the services schemata and some integrity constraints. According to the mappings the query is rewritten and the constraints as a query over the services which is transformed into several possible executable query plans considering possible limitations in accessing the services from which the content is extracted.

Framework consists of 3 layers:

Query formulation layer :- This layer allows users to convey their requests to the system by using an interface of global ontology which hides the specificity of the services. The main role of this layer is to rewrite the user query through mappings, whose results are expressed in the form of multi-domain conjunctive queries over physical services data with access limitations.

Query execution layer :- This layer generates a query plan optimized considering the parameter related to the services and the cost model. This optimization is carried out on issues,such as: (i) the types of operations involved in the query plan; (ii) available profiling information on specific services; (iii) ranking of the results.

Data layer :- The data layer addresses the view in the framework of the physical services; they may be either Web Services or wrapped, data-intensive Web sites. With estimates of the figures which are relevant to the optimization problem Services are constantly profiled so as to feed the optimizer of the layer above.

B. Query Optimization

The optimization approach consists in exploring a highly combinatorial solution space that characterizes all possible translations from the user query into fully instantiated query plan. It is separated into three phases, that giving details of query plans [2].

- The first phase is the selection of a given query rewriting such that every service is called with one of the available access patterns. This phase transforms the conjunctive query over web services into several annotated access queries over access patterns of the corresponding services.
- The second phase is the selection of a query plan for the given query rewriting. This phase fixes the order of execution of the query over the services as well as the position and kind of joins between services used in the plan.
- The third phase is the assignments of the exact number of fetches to be performed over the chunked services. This phase allows to fully determine the execution strategy for a query and therefore to compute its semantic score according to number of occurances of query keywords.

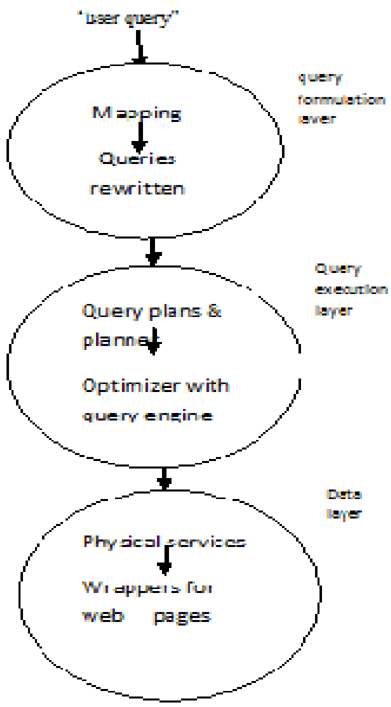


Fig.1, Reference Scenario[2]

C. Joiner Architecture

The focus is to develop techniques for integrating the results extracted from several existing search engines. Designing a system which offers a common interface to several, known search services [4] is our main idea. Architecture is given in fig.2-

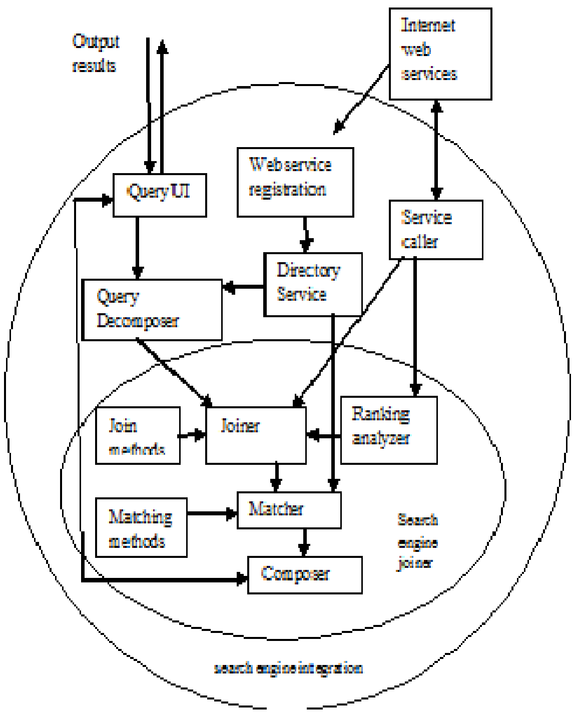
- The user interface takes in input queries from the user in the format of collections of terms and shows the results as output; allows users to indicate their.
- The query decomposer reduces the original query into many subqueries and recognises the services that can better address each subquery, using search services information coming from the search service directory.
- The joiner selects the join method, and when appropriate interacts with the service caller to request a new block of results from a search service.
- The matcher performs the join of the elements in a given tile, producing the result entries using correlation provided by the directory service.
- The composer builds the entry results which are sent to the user interface for their publishing.
- The ranking guesser is an optional module that estimates the average ranking of each block of

records which are given by a search from the service. The joiner activates the service caller to request Web services within our framework are registered by registration service.

D . Overall Architecture And Execution Flows

In the multi-domain query answering we analysed *two main flows: the **registration flow** - that deals with* the declaration and description of domains, and the registration of search services and their association to domains- and the ***query execution flow*** - that deals with the actual processing of the queries. Fig.3, shows the overall architecture of the system, together with the two main execution flows.

The terms shown by the activity flows are represented by a conceptual model that describes: (1) domains and their properties (classification taxonomies and associated concepts); (2)search services (request/response interfaces with annotations for in/out parameters and description of response , with functional and nonfunctional properties); (3) high level multi-domain user queries (simplified natural language queries, formed by subqueries); (4)low-level queries (adorned conjunctive datalog queries); (5)query plans (descriptions of strategies for query execution, by operations of coarse-granularity which consists with limitations to access and definining strategies which are ranking-aware for building results); and (6) query execution schedules (well-defined schedules of fine-granularity operations, including service invocations, which have the execution control flow) .



Fig,2, Joiner Architecture[4]

In registration flow, following problems are addressed: (a)semantic representation, storage, management, and access to domains and their descriptions; (b) semantic description, storage, management, and access to search services; (c) clustering of services based on similarity; (d) mapping of services to domains; and (e) definition of admissible join conditions between services.

In the query execution flow we address the following problems:(f) definition of proper interfaces for submission of multi-domain user queries; (g) splitting of the query into subqueries; (h) mapping of subqueries to domains; (i) mapping of subqueries on given domains to associated search services, for defining low level queries; (j) generation of query plans and their evaluation

against many cost metrics to select the most promising one for execution; (k) generation and processing of query execution plans; and (l) transformation and rendering of the results for user consumption.

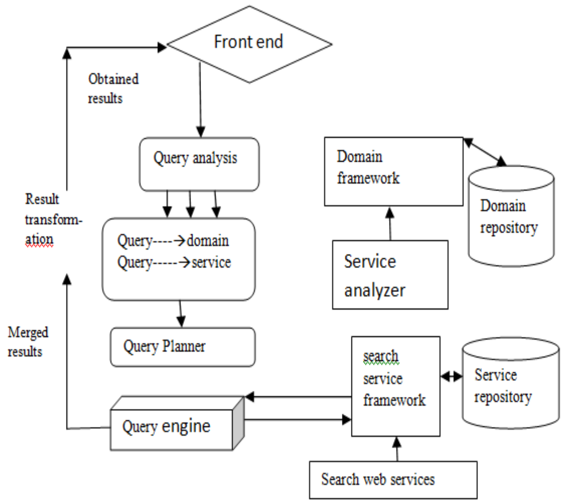


Fig.3, Overall architecture and execution flows[5]

E. Search Service

Searches may be of two types, web search (google) or web service search. Search services are to enable the annotation of the request/response interface of services. Here we are interested in those operations belonging to a Web service which perform data retrieval, and particularly in those operations that return itemized and ranked information. The service analyzer addresses the clustering of the available services, on seeing their similarity; the definition of join connections between services; and the mapping of services to domains.

F. Query to Domain And Service Mapping

To analyze the query, method consists of applying a first splitting of the sentence, and then to check whether the formed subqueries map consistently to separate domains, by invoking the Query-Domain mapper [5]. If incoherent mapping is we conjecture that the splitting is not well enough, and therefore we: (i) ask for feedback from the user; or (ii) try a different splitting based on cohesion of words w.r.t. domains. The final result of the splitting in (high-level) subqueries is therefore just a first step towards the mapping of subqueries to domains.

This component addresses the problems of mapping subqueries to domains and of mapping subqueries to associated search services, for defining low-level queries. The operation of mapping a query to a domain can be successful only if: (i) each subquery comprises only requests to one domain; and (ii) the words used in the subquery are unambiguous, thus allowing a crisp identification of their semantics (and therefore a correct mapping to the domains) [5].

Several ideas can be issued to optimize the recognition of query-subquery structures which comply with the separation into distinct domains of concern so as to achieve the objective (i); these include [5]:

- iterative invocation of the NLP tool based on defined lexical interpretation obtained from review from user, or review from other components;
- sustainment of annotations of search services or domains for assessing the correctness of the query splitting;
- On sentences syntax/logic analysis result is done.

G. *Semantic score and Relavnce and Reranking*

The number of occurances of any keyword of the query into the related web searches is Semantic score. Relevance is the semantic similarity between keywords and a specified web document. Our algorithm is different from other semantic similarity methods. We calculate the similarities between the keywords and each word in the document (ofcourse removing the stop words) to get the final relevance[8].

As we know, the search results are returned by search engines according to their importance and relevance. Generally the web pages which are most are returned at the upper positions, and attract much more attention from users.

III. CONCLUSION

This paper presented the set of terms need to be aaddressed when addressing multidomain queries. It gives the terms, architectures and methods to be involved. It encourages the use of web search and services to optimize the results resulting into relevant output. In our future work, we envision offering to users a more relevant and optimized multidomain query answering.

REFERENCES

[1] D. Braga, D. Calvanesey, A. Campi, S. Ceri_, F. Daniel, D. Martinenghi, P. Merialdoz, R. Torlonez, “NGS: a Framework for Multi-Domain Query Answerin,” .
[2] Daniele Braga, Stefano Ceri, Florian Daniel, Davide Martinenghi, “Optimization of MultiDomain Queries on the Web,” PVLDB '08, August 23-28, 2008, Auckland, New Zealand
[3] U. Srivastava, K. Munagala, J. Widom, and R. Motwani, “Query optimization over web services,” in VLDB-06, 2006, pp. 355–366.
[4] D. Braga, A. Campi, S. Ceri, and A. Raffio, “Joining the results of heterogeneous search engines,” Information Systems, Proceeding of the 32nd VLDB Conference, Seoul, Korea, 2006.
[5] Davide Barbieri, Alessandro Bozzon, Daniele Braga, Marco Brambilla, Alessandro Campi, Stefano Ceri, Emanuele Della Valle, Piero Fraternali,Michael Grossniklaus, Davide Martinenghi, Stefania Ronchi, Marco Tagliasacchi , “ Data-driven optimization of search service composition for answering multi-domain queries,” VLDB ’09, August 24-28, 2009, Lyon, France.
[6] Giorgio Ghisalberti, Marco Masseroli, Salvatore Vadacca,“ Multi-Domain Data Search and Retrieval:A Service-Oriented Life Science Scenario ”Dipartimento di Elettronica e Informazione, Politecnico di MilanoPiazza Leonardo da Vinci, 32, 20133 Milano, Italy.
[7] Ruofan Wang, Shan Jiang, Yan Zhang, “Re-ranking Search Results Using Semantic Similarity,” .
[8] S.G.Choudhary, S.R.Kalmegh, Dr. S. N. Deshmukh, “Semantic Search Algorithms based on Page Rank and Ontology: A Review,” 3rd International Conference on Intelligent Computational Systems (ICICS'2013) January 26-27, 2013 Hong Kong (China)