Scientific Journal Impact Factor (SJIF): 1.711

IJMTER

International Journal of Modern Trends in Engineering and Research

ISSN (ONLINE) : 2349 - 9745 ISSN (PRINT) : 2393 - 8161

Ranking and Recommendation of Query Form: A Survey

A. R. Gaikwad[1], C. J. Awati[2]

[1]*Department of Technology, Shivaji University, Kolhapur*

[2]*Department of Technology, Shivaji University, Kolhapur*

Abstract— In database research, there has been some scattered work with the programmed extraction of comparability/positioning capacities from a database. Relational database system supports the effective execution of complex queries. During Query Recommendations users employ query interface to issue a series of SQL queries in order to analyze the data and mine for interesting data. The ranking mechanism depends on captured user preferences. Whereas, the aim of Dynamic Query Form is to catch a users preference and rank query form components insisting him/her to take decision. This paper reviews dynamic query generation.

Keywords— Query Form, Query processing, Ranking, User Interaction, Query Form generation.

I. INTRODUCTION

Query form is amongst the most generally utilized client interfaces for querying databases. Conventional query forms are composed and predefined by designers or DBA's in different information management systems. With the quick advancement of web data and exploratory databases, modern databases turn out to be vast and complex.

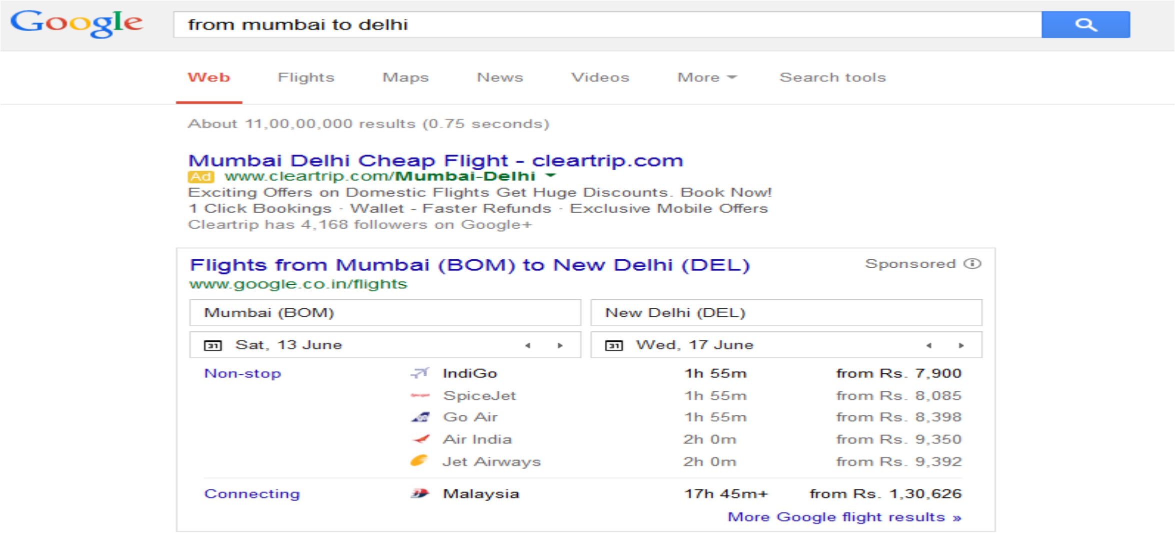
A Dynamic query form (DQF) [1] system which generates the query forms according to the user's expectation at the execution time has been proposed. The system provides a remedy for the query interface in huge and critical databases. During the interaction between the users for the Query Form Enrichment DQF involves recommending ranked list of the query form components to the user and the user selects the desired form components into the current query form. Whereas, during Query execution, DQF executes the query and displays the result and the user fills the current query form, submit a query and provides the feedback regarding the query results.

Automated ranking of the aftereffects of a question is prevalent parts of the inquiry demonstrate in Information Retrieval (IR) that we have created to depend on upon. Interestingly, database frameworks reinforce just a Boolean query model. For instance, in SQL database a selection query gives back all tuples that satisfy the conditions in the query. Regardless of the possibility that we get the positioning capacities right, for huge databases, we need to minimize their effect on query preparing. Although transformed records are prevalent data structures for effective recovery in IR, they are lacking for our reasons as we look for uncertain matches including downright and numerical qualities.

A system has been built in which several ranking algorithms have been implemented on a relational DBMS. The system consists of two important components firstly a pre-processing component and secondly a query processing component. The pre-processing segment is a ranking function extractor that influences information and workload qualities. The query processing component is a Top-K algorithm that utilizes the ranking capacity and adventures the physical database outline. A typical feedback of database frameworks is that they are difficult to inquiry for clients uncomfortable with a formal question dialect. To address this issue, structure based interfaces and watchword pursuit have been proposed.

As the achievement of internet searchers makes liberally clear, when confronted with finding reports of interest, the overall public is effective at utilizing keyword search to fulfil the assignment.

The approach of keyword search prompting a structure has as of now been utilized as a part of specially appointed way via web search tools, for example Google and Yahoo.



from mumbai to delhi

About 11,00,00,000 results (0.75 seconds)

Mumbai Delhi Cheap Flight - cleartrip.com
Ad www.cleartrip.com/Mumbai-Delhi
Exciting Offers on Domestic Flights Get Huge Discounts. Book Now!
1 Click Bookings · Wallet - Faster Refunds · Exclusive Mobile Offers
Cleartrip has 4,168 followers on Google+

Flights from Mumbai (BOM) to New Delhi (DEL)
www.google.co.in/flights

Mumbai (BOM)		New Delhi (DEL)	
Sat, 13 June		Wed, 17 June	
Non-stop	IndiGo	1h 55m	from Rs. 7,900
	SpiceJet	1h 55m	from Rs. 8,085
	Go Air	1h 55m	from Rs. 8,398
	Air India	2h 0m	from Rs. 9,350
	Jet Airways	2h 0m	from Rs. 9,392
Connecting	Malaysia	17h 45m+	from Rs. 1,30,626

More Google flight results »

Figure 1: For the query from "Mumbai to Delhi", Google returns a simple form for finding flight tickets besides links to website

As indicated in Figure 1, querying Google with "from Mumbai to Delhi" raises an inquiry interface to purchase plane tickets as the first result. Be that as it may, on the Web, the essential task of magic word inquiry is to lead clients investigating the Web to reports significant to their search. While the web index might infrequently give back a structure significant to an inquiry, there is no longing to bolster an extensive variety of conceivable organized questions. By contrast, we try to add to an exhaustive methodology that permits clients to answer a wide variety of questions over a single organized information set.

II. LITURATURE REVIEW

Ranking is an important component in collaborative filtering research [3]. The collaborative filtering methods require training data and also their ranked results using queries. In contrast, we need workloads consisting of only queries. In database research, there has been some scattered work with the programmed extraction of comparability/positioning capacities from a database.

The early work of [5] considered vague/imprecise similarity-based querying of databases. The problem of integrating databases and information retrieval systems has been attempted in several works [6, 7, 8, 9]. Information retrieval based approaches have been extended to XML retrieval in [10]. A keyword-based retrieval system over databases is proposed in [4].

QBE (Query-By-Example) [11] is language for non-programmers to query and update a relational database. The users are provided with Skeleton tables of a database or customized forms so that they can fill in the blanks with "examples" to determine query constraints. In spite of the fact that much less difficult than SQL, despite everything they oblige a comprehension of the social model, and could give clients inconvenience when the composition is entangled. These works don't consider the issue of looking over an arrangement of structures, probably because only few forms are used.

Jayapandian et al. described an approach that automatically generates forms for a database based on a sample query workload [12], and more recently, an approach to automatically create a form-based interface, with the goal of maximizing expressivity while respecting specified bounds on interface complexity [13]. Since their objective is to make a little arrangement of frames, they additionally don't consider the issue of browsing a set of structures.

An alternative to writing structured queries is keyword search over databases [4, 14, 15] however; it has limited ability to exploit structured data. For instance, totals, projections, range questions, and inquiries that determine which characteristic must contain a wanted consistent, are all outside the extent of "fundamental" decisive word look. Liu et al. [15] proposed to automatically distinguish between schema terms and value terms in a keyword query, and adopted a new ranking strategy for handling keyword queries with schema terms. Contrasted with utilizing structures, this methodology has little backing for organized questions. To be sure, a noteworthy inspiration for our methodology is to permit clients who would prefer not to utilize SQL to still have the capacity to influence the benefit of questioning organized information.

So far, the work that has been done in the area of personalized databases has focused to keyword-based query recommendation systems [16]. In this circumstance, a customer can take up with a social database through a web interface that allows him/her to submit conclusive words and recoup pertinent substance. The personalization procedure relies on upon the client's keyword queries, those of past customers, and furthermore an express customers profile that records the customer's slants concerning the substance of the databases. Hence, this approach is different from this scenario in various ways. Initially, the proposed framework is intended to help clients who posture complex SQL questions to relational databases. Additionally, the framework does not require from its clients to make an unequivocal profile. This gives a larger amount of adaptability to the system, since the same client may have diverse data needs during distinctive investigations of the database.

The challenges of applying data mining techniques to the database query logs are also addressed in [17]. In this work, the creators diagram the structural engineering of a Collaborative Query Management System focused everywhere scale, shared-information situations. As a major aspect of this structural planning, they freely propose that information mining strategies, for example, grouping or affiliation guidelines can be connected to the inquiry sign so as to furnish the clients with question suggestions. We ought to push, nonetheless, that in spite of our work, the creators don't give any specialized subtle elements on how such a proposal framework could be actualized.

III. MODULES

This section consists of two mechanisms related to query forms namely customized query form and ranking mechanism. They are explained in detail as follows:

3.1. Customized Query Form.

Visual interfaces for designers with a specific end goal to make or modify inquiry structures have been given. The issue with respect to those devices is that, they are accommodated the expert engineers who are acquainted with their databases, rather than the end-clients. It proposed a framework which permits the end-clients to modify the present inquiry structure at the execution time. In this way, an end-client may not be acquainted with the database. On the off chance that the database mapping is exceptionally colossal, it is troublesome for them to find suitable database elements and attributes and to create the craved question frames.

3.2. Ranking Mechanism.

Query forms are generated to return the users expected result. There are two types of measures to calculate the nature of query which are: precision and recall [2]. Query forms are designed to return the users desired output. To fulfill the expected precision and expected recall we can imagine the execution of the query form, query forms are fit for give the definite queries by sorted out the inputs and that particular query definitely fulfill the precision and recall. Actually the essential level of the sales results gives the expected precision, which is intrigued by the user. In other side the predictable level of the user’s delighted information cases returned from the present query forms results in the expected recall. The client loosening up movement is laid out in association of the client's look at on sales results exhibited by the query form. A substantial case, if some information events are clicked by the customer, these information cases must have high customer side hobbies. At that point, the query form components which can catch these information occurrences need to be ranked higher than other elements.

The ranking model is in view of two thoughts, for example, client closeness and query similarity. Client closeness demonstrates that multiple clients can have same inclinations. Query closeness shows that diverse clients can have comparable questions. With a specific end goal to perform this ranking of clients and queries are to be kept up. A workload record that incorporates the client and query ranking cut off points has been conveyed. The customer managed the database when another record gone into the database. There are possibly different clients who went on that query leading up to now and there may be unclear queries went on starting at this point. The proposed model has two cases blended. They are known as client addictive ranking model and query addictive ranking model.

III. CONCLUSION

The main objective of the review paper is to focus on dynamic query generation. We also discussed the customized query forms and ranking mechanism. Different authors presented different techniques for ranking and querying. The issues that arise in the implementation: designing and creating forms in a systematic fashion, handling keyword queries that are a blend of information terms and composition terms filtering through structures that would deliver no outcomes regarding a user's query and positioning and showing structures in a manner that help users discover helpful structures all the more rapidly. Thus, a survey of dynamic query form generation approach which helps the user to dynamically create the query forms.

REFERENCES

[1] L. Tang, T. Li, Y. Jiang and Z. Chen, “Dynamic Query Forms for Database Queries” IEEE Trans. Knowl. Data Eng.,vol. 26, No. 9, pp. 2166-2178, Sept. 2014.

[2] G. Salton and M. McGill, Introduction to Modern Information Retrieval. New York, NY, USA: McGraw-Hill, 1984.

[3] J. Breese, D. Heckerman and C. Kadie. Empirical Analysis of Predictive Algorithms for Collaborative Filtering. 14th Conference on Uncertainty in Artificial Intelligence,1998.

[4] S. Agrawal, S. Chaudhuri and G. Das. DBXplorer: A System for Keyword Based Search over Relational Databases. ICDE 2002.

[5] A. Motro. VAGUE: A User Interface to Relational Databases that Permits Vague Queries. 187- 214, TOIS 6(3) 1988.

[6] W. Cohen. Integration of Heterogeneous Databases Without Common Domains Using Queries Based on Textual Similarity. SIGMOD, 1998.

[7] W. Cohen. Providing Database-like Access to the Web Using Queries Based on Textual Similarity. SIGMOD 1998.

[8] N. Fuhr. A Probabilistic Framework for Vague Queries and Imprecise Information in Databases. VLDB 1990.

[9] N. Fuhr. A Probabilistic Relational Model for the Integration of IR and Databases. ACM SIGIR Conference on Research and Development in Information Retrieval, 1993.

[10] A. Theobald and G. Weikum. The Index-Based XXL Search Engine for Querying XML Data with Relevance Ranking. 477-495, EDBT 2002.

[11] M.M. Zloof. Query-by-Example: the Invocation and Definition of Tables and Forms. VLDB 1975.

[12] M. Jayapandian, H. V. Jagadish. Automating the Design and Construction of Query Forms. ICDE 2006.

[13] M. Jayapandian, H. V. Jagadish. Automated Creation of a Form-based Database Query Interface. VLDB 2008.

[14] G. Bhalotia, A. Hulgeri, C. Nakhe, S. Chakrabarti, and S. Sudarshan. Keyword Searching and Browsing in Databases using BANKS. ICDE, 2002.

[15] F. Liu, C. Yu, W. Meng, A. Chowdhury. Effective Keyword Search in Relational Databases. SIGMOD 2006.

[16] Koutrika, G., Ioannidis, Y.: Personalized queries under a generalized preference model. In: ICDE 2005: Proceedings of the 21st International Conference on Data Engineering, pp. 841–852 (2005).

[17] Khoussainova, N., Balazinska, M., Gatterbauer, W., Kwon, Y., Suciu, D.: A case for a collaborative query anagement system. In: CIDR 2009: Proceedings of the 4th biennal Conference on Innovative Data Systems (2009).