Volume 4, Issue 11, November 2014 ISSN: 2277 128X

International Journal of Advanced Research in Computer Science and Software Engineering

Research Paper

Available online at: www.ijarcsse.com

A Review on Dynamic Query Forms for Database Queries

Priyanka P. Nikam
Computer Engg. Department,
MCOERC Nashik Pune University, India

Abstract— With rapid development of scientific databases and internet information databases are becoming very huge in size and complex in nature. These databases maintain large and heterogeneous data, with large number of relations and attributes. So it is very difficult to design a set of static question forms to answer various ad-hoc database queries on these modern databases. Thus there is need of such system which generate Query Forms dynamically according to the user's need at run time. The proposed system Dynamic Query Form i.e.DQF system going to provide a solution by the query interface in large and complex databases. In proposed system, the core concept is to capture user interests throughout user interactions and to adapt the question type iteratively. Every iteration consists of 2 sorts of user interactions: Query Form Enrichment and Query Execution. In Query Form Enrichment DQF would recommend a ranked list of query form component to user so he/she can select desired form components into current query form. In Query Execution user fills current query form and submit query, DQF going to show result and take feedback from user on provided query results. A user would have facility to fill the query form and submit queries to view the query result at each iteration. So that a query form could be dynamically refined till the user satisfies with the query results.

Keywords— Query Form, Query Execution, User Interaction,Iteration.

I. INTRODUCTION

A database is only as functional as query interface allows it to be. If a user is not capable to communicate to the database what he or she wishes from it, even the richest data store provides petite or no value. Writing well-structured queries, in languages such as SQL and XQuery, can be challenging due to a number of reasons, including the user's lack of familiarity with the query language and the user's ignorance of the underlying schema. A form-based query interface, which only requires filling blanks to identify query parameters, is precious since it helps make data users with no knowledge of official query languages or the database schema. In practice, form-based interfaces are used frequently, but usually each form is designed in an adhoc way and its applicability is restricted to a small set of fixed queries. Query form is one of the majority used user interfaces for querying databases. Traditional query forms are designed and predefined by developers or DBA in various information management systems. With the rapid development of web information and scientific databases, modern databases become very large and complex.

Dynamic question type system: DQF, a question interface that is capable of dynamically generating question forms for users. Different from ancient document retrieval, users in information retrieval area unit usually willing to perform several rounds of actions (i.e., refinement question conditions) before distinctive the final candidates. The essence of DQF is to capture user interests throughout user interactions and to adapt the question type iteratively. Every iteration consists of 2 sorts of user interactions: it contains only a few primary attributes of the information. The essential question type is then enriched iteratively via the interactions between the user and our system till the user is satisfy with the question results.

Goal of this paper is to show that the advantages of using dynamic query forms for database over the existing static query forms.

II. LITERATURE SURVEY

A lot of research works focus on database interfaces which assist users to query the relational database without SQL. QBE (Query-By-Example) [6] and Query Form are two most widely used database querying interfaces. Current studies and works mainly focus on how to generate the query forms

Modified Query Form: The tools provided by the database clients make great efforts to help developers generate the query forms, such as Easy Query [2], Cold Fusion [1] and so on. They provide visual interfaces for developers to create or customize query forms. The problem of those tools is that, they are provided for the professional developers [3].H.V. Jagadish proposed a system which allows end-users to customize the existing query form at run time [7]. If the database schema is very large, it is difficult for end user to find appropriate database entities and attributes

Automated creation of forms: M. Jayapandian presented a data-driven method [3]. It first finds a set of data attributes, which are most likely queried based on the database schema and data instances. Then, the query forms are generated based on the selected attributes.

Automating the design and construction of query forms: H.V. Jagadish presented a workload-driven method

[8].It applies clustering algorithm on historical queries to find the representative queries. The query forms are then generated based on those representative queries. One problem of the aforementioned approaches [3],[8] is that, if we generate lots of query forms in advance, there are still user queries that cannot be satisfied by any one of query forms. Another problem is that, when we generate a large number of query forms, how to let users find an appropriate query form would be challenging.

Combining keyword search and forms: A solution for aforementioned approaches [3], [8] is proposed in [9].It automatically generates a lot of query forms in advance. The user inputs several keywords to find relevant query forms from a large number of pre-generated query forms but it is not appropriate when the user does not have concrete keywords to describe the queries

III. SYSTEM OVERVIEW

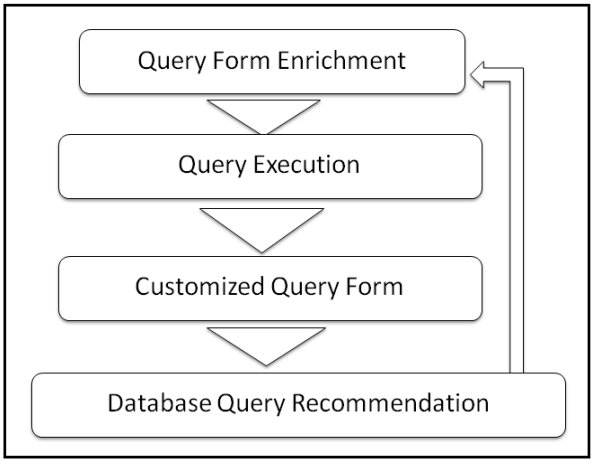

Fig.1 System Modules

Fig 1 shows the flow of the system modules. There are four basic modules in DQF system are :-

A. Query Form Enhancement

In this module DQF recommends a ranked list of query form components to the user. So that the user selects the desired form components into the current query form.

B. Query Execution

Firstly the user fills out the current query form and submit a query. Then DQF executes the query and shows the results.

C. Customized Query Form

They provide visual interfaces for developers to create or customize query forms. The problem of those tools is that, they are provided for the professional developers who are familiar with their databases, not for end-users. If proposed a system which allows end-users to customize the existing query form at run time. However, an end-user may not be familiar with the database. If the database schema is very large, it is difficult for them to find appropriate database entities and attributes and to create desired query forms.

D. Database Query Recommendation

Recent studies introduce collaborative approaches to recommend database query components for database exploration. They treat SQL queries as items in the collaborative filtering approach, and recommend similar queries to related users.

IV. CONCLUSIONS

Query interfaces play a vital role in determining the usefulness of a database. A form-based interface is widely regarded as the most user-friendly querying method. In this paper, we have developed mechanisms to overcome the challenges that limit the usefulness of forms, namely their restrictive nature. In this paper we propose an interactive query form generation approach which helps users to dynamically generate query forms. As future work, we will study how our approach can be extended to non-relational data. As for the future work, we plan to develop multiple methods to capture the user's interest for the queries besides the click feedback. For instance, we can add a text-box for users to input some keywords queries.

REFERENCES

[1] ColdFusion. http://www.adobe.com/products/coldfusion/.
[2] EasyQuery. http://devtools.korzh.com/eq/dotnet/.
[3] M. Jayapandian and H. V. Jagadish. Automated creation of a forms-based database query interface. In Proceedings of the VLDB Endowment, pages 695–709, August 2008.
[4] S. Agrawal, S. Chaudhuri, G. Das, and A. Gionis. Automated ranking of database query results. In CIDR, 2003.

[5] G. Salton and M. McGill. Introduction to Modern Information Retrieval. McGraw-Hill, 1984.

[6] M. M. Zloof. Query-by-example: the invocation and definition of tables and forms. In Proceedings of VLDB, pages 1–14, Framingham, Massachusetts, USA, September 1975.

[7] M. Jayapandian and H. V. Jagadish. Expressive query specification through form customization. In Proceedings of International Conference on Extending Database Technology (EDBT), pages 416–427, Nantes, France, March 2008

[8] M. Jayapandian and H. V. Jagadish. Automating the design and construction of query forms. IEEE TKDE, 21(10):1389– 1402, 2009.

[9] Liang Tang, Tao Li, Yexi Jiang, and Zhiyuan Chen, "Dynamic Query Forms for Database Queries," IEEE Trans. On Knowledge and Data Engg. Vol:PP No:99 Year 2013

