ISSN: 2321-7782 (Online)
Impact Factor: 6.047

Volume 4, Issue 5, May 2016

International Journal of Advance Research in Computer Science and Management Studies

Research Article / Survey Paper / Case Study

Available online at: www.ijarcsms.com

A Review on Generation of Dynamic Query Forms based on User Preferences

Rekha Nikam[1]
PG Student, Computer Engineering Department,
MET'S IOE
Nashik, Maharashtra – India

Prof. V. B. More[2]
Asstt. Professor, Computer Engineering Department,
MET'S IOE.
Nashik, Maharashtra – India

Abstract: The query forms are provides the interface between user and databases. Different Query forms are designed for satisfy the different user needs. As the time passes the demands of the users can be changed according to requirement at that time. Traditional static query form does not satisfy the different ad-hoc queries of user interest on databases. The growth in internet technology leads to internet data very large containing thousands of attributes and hundreds of relation. To retrieve required information from such complex databases it is very headache for the user who doesn't know anything about the database schema or how the databases internal structure has created. The solution to this problem is Dynamic query form technique [1] which is used for dealing with such modern large databases. User can modify form components according to their requirement, these changes are used to generate new form and hence Query form is generated dynamically. DQF (dynamic query form) uses two level of user interface: Query Form Enrichment and Query Execution. In DQF user can modify query form iteratively and at each iterations the user can select form component from the list and add to the form. This process is continuously repeated until the user is satisfied by the query result. We introduce another feature called Preference card.

Keywords: Dynamic Query Form, Query Form Enrichment, Query Execution, Preference card.

I. INTRODUCTION

The 'Query form' is providing a way to deal with the database. By using query form user can retrieve required information from the databases. The traditional query forms designed and predefined by the developers and literate of DBA's in various information management systems such as scientific databases, business application database etc.

As there is rapid development in the web informatics and scientific databases, the databases become very huge and complicated. For instance the database of product manufacturing company which contains the information about hundreds of products entities. It is hard to structure a set of static query forms to satisfy different ad-hoc database queries for complex web databases. Various database management tools are used for development of customized queries on databases. But the development of customized queries totally based upon manual editing of the form. If the user does not familiar with databases schema then they may confused by large number of attributes of complex databases. It is difficult to use relational database for person which not having the knowledge of databases manipulation. To provide facility for non-technical user to query the relational databases without using SQL, DQF mainly focused on creation of user friendly database interfaces. Dynamic Query Form system can be used for preparing a query interface which is capable of dynamically producing query forms for the users.

The query forms are modified at each iteration. Each iteration is gone through two phases of user interactions:

1) Query Execution, and

2) Query Form Enrichment.

The working of the DQF starts with a basic query form containing very few primary attributes of the database. It is then modify iteratively by the means of interactions between the user and the system, unless the user is satisfy with the query outcome. Dynamic query form system used for creating the query forms according to the run time desire of the user. The technique uses F-score determine the goodness of a query form. Using F-score system can rank and recommend the probable query form components, so that the users can filter the query form easily. The metric is used for approximate the goodness of the query form and collection of variety of form components, It is done by efficient algorithms. The DQF is an online tool so users usually expect quick response therefore efficiency is very important. Preference card is contains the forms which are best rated by the user. Preference card technique helps user to segregate the best forms which can be needed by user in future.

II. Literature Survey

M. Jayapandian and H. V. Jagadish proposed [3],[4] automatic approaches to generate the database query forms without user intervention. It first finds a set of data attributes, which are queried based on the database schema and data instances. Then, the query forms are generated based on the user selected attributes. Workload-driven method applies clustering algorithm on historical queries to find the similar type of queries. Query forms are generated based on those representative database queries. The user inputs keywords to find relevant query forms from a large number of already generated query forms. It works better in the databases which have rich textual information in data tuples and schemas. But in some cases it is not appropriate when the user does not have fixed keywords to describe the queries.

Ideal user interfaces have been developed by A. Nandi and H. V. Jagadish **[2]** to guide the user to type the database queries based on the query workload, the data distribution and the database schema. The queries in their work are in the forms of SQL and keywords. They provide instance response interface. The user is presented with a text box, and is expected to type in keywords, a simple interaction type that is considered the standard in search systems. As soon as the user starts typing text into the textbox, the system instantly begins to provide suggestions. There are two sets of suggestions: 1) first set includes ranked list of keys, or entity names. This allows the user to discover the schema as an ongoing implicit process. 2) The second set of suggestions is a ranked list of text fragments from the data. Each text fragment is expanded to a full key: value pair on selection. This allows the user to differentiate between text values based on their type. Clicking on the icons in the textbox will wrap the text selection around the associated key name and hence trigger the drop down to modify the key. The estimated number of tuples, or result objects from the resultant constructed query is provided on the right of each suggestion.

Query refinement is proposed by W. B. Frakes and R. A. Baeza-Yates **[9]** used by most information retrieval systems. It recommends new terms related to the query or modifies the terms according to the navigation path of the user in the search engine. A database query is a structured relational query, not a set of keywords.

S. B. Roy, H. Wang, U. Nambiar, G. Das, and M. K. Mohania **[10]** evolved Dynamic faceted search is a type of search engines where relevant facets are presented for the users according to user navigation paths. Dynamic faceted search engines are similar to dynamic query forms if and only if consider *Selection* components in a query. A database query form has other important components, called as *Projection* components. *Projection* components control the output of the query form and it is mandatory. *Selection* and *Projection* have inherent influences to each other.

G. Chatzopoulou, M. Eirinaki, and N. Polyzotis **[11]** introduce collaborative filtering approaches to recommend database query components for database exploration. In this SQL queries are considered as item. By using collaborative filtering the similar type of queries are clustered into one group and recommend similar queries to related users. In this approach goodness of the query results are not considered. The difference between DQF and database query recommendation approach is that, their recommendation is a complete query and DQF recommendation is a query component for each iteration.

K. Chen, H. Chen, N. Conway, J. M. Hellerstein, and T. S. Parikh **[12]** developed an adaptive forms system for data entry, which can be dynamically changed according to the previous data input by the user. In this data quality can be maintain at every stage of the data life-cycle, from creation to storing in the database. They mainly focus on improving the quality of manually entered data.

USHER's provides the facilities as follows:

1) Automatically extract training data from a Microsoft Access database;

2) Automatically refining of designed form; and

3) Execute the forms with and without smart formatting and quality assurance. DQF is different from other query form as it is dealing with database query forms instead of data-entry forms.

III. Processing Strategy

A dynamic query form system which generates the query forms according to the user's need at run time. The system provides a solution for the query interface in large and complex databases. This provides DQF, a novel database query form interface, which is able to dynamically generate query forms. The importance of DQF is to capture a user's preference and rank query form components, assisting user to make decisions. The generation of a query form is an iterative process and is guided by the user. At each iteration, the system automatically generates ranking lists of forms. The user can add the desired form components into the query form. The ranking of form components is based on the captured user preference. A user can also fill the query form and submit queries to view the query result. In this way, a query form could be dynamically refined till the user satisfies with the query results.

A. System Modules

DQF contain following module:

1) Query Form Enrichment:

User selects the form component from the component list and adds to the form.

2) Query Execution:

First the user fills out current query form and submits a query. Then DQF executes the query and shows results.

1) Query Enrichment contain following component:

- Query forms

The query form is defined in this section. Each of the query form represented as a SQL query template. As ad-hoc join is not handled in dynamic query forms approach.

2) Query Execution contain following component:

- Query Results

When the user submits the query form the output is given by the system. If the user is not satisfied by the result then user can select form component from the component list and fill the form and submit it. It is iterative process.

IV. Overall Process Analysis

In DQF i.e. dynamic query form system the dynamic form generation process is going through the two phases: 1) Query Form Enrichment and 2) Query Execution. The process is start with the simple form. In query form enrichment phase user enrich the form by adding desire form component to the current form. For that there is one form component list provided to the user. From that form component list user can select the form component. The selected form components are then added to the form. User can give conditions to the more than one form component to get the desire result. The user can give rating to the forms as Best, Average, Under Average and Bad. If the user is not satisfied with the query result then user can do same process iteratively.

Ranking of the form can increase the efficiency and effectiveness of the system. There are two types of ranking methods: baseline method and random method. The baseline method ranks projection and selection attribute in ascending order of their schema distance of current query form. In DQF approach for ranking projection component ranking score method is used, it is obtain by comparing the computed ranking with the optimal ranking [1]. F- measure is used for measuring ranking selection form component [1]. By using two separate methods of ranking, the ranking process became more accurate and effective.

V. Conclusion

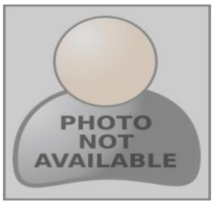
Dynamic query form approach helps users to dynamically generate the query forms. Dynamic query form generation method includes two phases: Query Execution and Query Form Enrichment, due to these two phases of interactions DQF system becomes more effective than customize query form technique. The highly rated forms are shown in the separate list to help the user to choose the correct form which can give desire output from the number of forms. The ranking is based on the forms output and given by user. User preferences are captured by runtime feedback given by user. Dynamic query form system is the online system therefore it has to be more accurate and give quick response to user.

References

1. 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.
2. M. Jayapandian and H. V. Jagadish. Automating the design and construction of query forms. IEEE TKDE, 21(10):1389– 1402,2009.
3. M. Jayapandian and H. V. Jagadish. Automated creation of a formsbased 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. A. Nandi and H. V. Jagadish. Assisted querying using instantresponse interfaces. In Proceedings of ACM SIGMOD, pages 1156–1158, 2007.
9. W. B. Frakes and R. A. Baeza-Yates. Information Retrieval: Data Structures and Algorithms. Prentice-Hall, 1992.
10. S. B. Roy, H. Wang, U. Nambiar, G. Das, and M. K. Mohania. Dynacet: Building dynamic faceted search systems over databases. In Proceedings of ICDE, pages 1463–1466, Shanghai, China, March 2009.
11. G. Chatzopoulou, M. Eirinaki, and N. Polyzotis. Query recommendations for interactive database exploration. In Proceedings of SSDBM, pages 3–18, New Orleans, LA, USA, June 2009.
12. K. Chen, H. Chen, N. Conway, J. M. Hellerstein, and T. S.Parikh. Usher: Improving data quality with dynamic forms. In Proceedings of ICDE conference, pages 321–332, Long Beach, California, USA, March 2010.

AUTHOR(S) PROFILE

Rekha Nikam, Received degree in Information Technology from the MGM's JNEC, Aurangabad. Currently pursuing master degree in computer engineering from MET's IOE, Nashik. The domain of interest is data mining.



Prof. V. B. More, Presently he is working at MET BKC Institute of Engineering, Nashik, Maharashtra as Assistant Professor. He has presented and published papers at national and international conference as well in journals on various aspects of the computer engineering. He received the master degree in computer engineering (Automation) from RGPV Bhopal, SGSITS Indore MP. His research interest areas are networking and data mining and information retrieval.