

Dynamic Query Form Generation

Kalyani L. Jaiswal

Bachelor Student, Computer Science & Engineering, DES'COET Dhamangaon Rly, India.

Sarika S. Joshi

Bachelor Student, Computer Science & Engineering, DES'COET Dhamangaon Rly, India.

Abstract – There is rapid enhancement in web services and web popularity, as the web users are increases day by day. Database is collection of number of relations and attributes which makes it more complex. Such a large and heterogeneous data is difficult to be maintained. By this enhancement of web services and scientific databases, it is not able to get user required result by static query forms. Developers are familiar with the queries but might face problem of writing queries again and again for different situations. And problem is also faced by those who are not familiar with databases but needs to access data from databases. To solve above problem we are proposing a model called Dynamic Query Form which will be able to dynamically generate query form. It will provide recommendations of components based on user preferences. It will not just generate dynamic query form but new data can be added and deleted by user accordingly. User adds the desired components, specifies the conditions and gets the result. The query form is iterative. The ultimate goal of DQF is user's satisfaction.

Index Terms – User preference, query form generation, recommendations, enrichment, DQF (dynamic query form), SQF (static query form), CQF (customized query form), query execution.

1. INTRODUCTION

Query form generation is iterative process and is guided by the users. Traditional databases are static and designed by professional to maintain information [1]. With the quick

improvement, present day databases turn out to be vast and complex. Many web databases, for example, Freebase and DBPedia, commonly have a huge number of organized web substances [4] [2]. In this way, it is hard to plan static inquiry structures to fulfill different specially appointed database questions on those unpredictable databases. It is conceivable that many questions are not characterized in it but rather is required by client.

Many existing database administration and improvement apparatuses, for example, EasyQuery [3], Cold Fusion [5], SAP and Microsoft Access, give a few systems to give clients a chance to make modified questions on databases. It is valuable for engineers or for the individuals who know about Databases. On the off chance that a client is not acquainted with the database blueprint ahead of time, those hundreds or a large number of information qualities would confound him/her without a doubt.

In this, we propose a Dynamic Query Form framework: DQF,

an inquiry interface which is able to do progressively producing question shapes for client's benefit.

Query Form Execution	<ol style="list-style-type: none"> 1. User fills the query form using query form components. 2. User specifies the desired conditions. 3. DQF executes the query and displays the result. 4. User provides feedback about the result.
Query Form Enrichment	<ol style="list-style-type: none"> 1. User is provided recommendation ranked list of query form components. 2. User adds or removes the components according to his desires.

Every emphasis comprises of two sorts of client associations: Query Form Execution and Query Enrichment (see Table 1). Figure 1 demonstrates the work-stream of DQF. In this system, database contain the very few attributes which is use to form basic query. Formation of basic query is first step. The client fill the query, generate the query and the result. The process of formation of basic query is continue until the client is satisfied with the query result and process is end otherwise the client is again fill the query these process is continuous.

Next, the client give criticism whether the client is happy with the question frame and in the event that he is not fulfilled then there is an alternative of inquiry improvement through which client can include or evacuate parts as he cravings and prescribed positioned rundown will be given to help client to taking choices. In the event that client is fulfilled, our objective is finished generally this procedure proceeds until client is completely fulfilled.

2. RELATED WORK

DQF is most widely used user interface for querying the database but before DQF there were lots of research performed for finding an interface that makes user capable of querying database. The main motive of DQF is to adapt itself according to user satisfaction. Following researches were carried out before the evolution of DQF:

2.1. Static Query Form:

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. It is a workload-driven method. One problem of the aforementioned approaches is that, if the database schema is large and complex, user queries could be quite diverse. In that case, even 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 and desired query form would be challenging.

2.1. Customized Query Form:

For creation of customized query they provide the visual interface for the developer. This query form is not easy for non-technical client because client is not familiar with their database. It is only used full for professional developer who are familiar with their database, that is who are not familiar with database for large and complex database user select the attribute present in the database and create desired query forms.

3. PROPOSED MODEL OF SYSTEM

For an implementation of proposed system is as following:

1. Designing of Query form
2. Adding Desired Components to Form
3. Applying Desired Conditions
4. Dynamic form generation
5. Query execution
6. Query Enrichment

1. Designing of Query form

Components can be added or deleted by user in this module. This module will ask for table name and its attributes but as user is not familiar with database system will ask like about what you want to save information and as user will type the name, the table of that name will be created then what kind of information you want to save the user will fill attributes and at last it will ask for values of those attributes.

2. Adding Desired Components to Form

User first enters name of database and gets a list of tables that are present in that particular database. Each table consists of components present within it and user selects various components which user needs.

3. Applying Desired Conditions

4. RESULTS AND DISCUSSIONS

This paper states that a DQF i.e., Dynamic Query Form generation approach which helps users dynamically generate

After choosing desired components user needs to specify conditions. User selects the component on which the condition is to be applied. Next the user selects the condition to be applied and at last user puts the value of condition.

4. Dynamic form generation

After applying desired conditions the dynamic form is generated where the user will fill the required fields and submit it.

5. Query Execution

After submitting form the DQF generates query for given data and presents the result. The user will either be satisfied with the result and will end up with that result otherwise if not satisfied he will perform query enrichment.

6. Query Enrichment

If user is not satisfied with the result he can choose query enrichment through which the user can choose or remove components as per requirement and can alter the specified condition and will execute the query again. At each iteration the user will get a recommended list of components which will help user in making decisions.

Recommendations are presented on the basis of user preferences. The components selected maximum by user will be ranked accordingly.

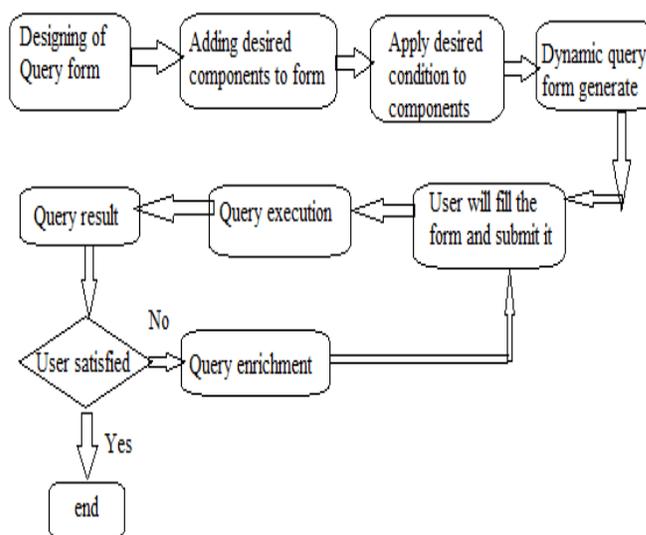


Fig 1. Workflow of system

query forms. The system is intended to generate different types of queries on the basis of user feedback with iterations of the query enrichment process. A user can also fill the query form

and submit queries to analysis the query result at each iteration. The dynamic query form generation system focuses on providing user friendly GUI, efficient and fast query generation, and query enrichment.

5. CONCLUSION

Query interfaces play a most important role in determining the usefulness of a database. User-friendly querying method is provided by form-based interface that are widely used in large and complex database. In the proposed system user interact with the system and form the dynamic query as per requirement. The proposed system provides higher success rate and simple query for user. We capture user feedback at runtime by click through method. As future work, In future proposed system works with non-relational data. In the future we plan to develop multiple methods to capture the users interested for the queries besides the click feedback. For

instance, we can add a text-box for users to write some input keywords queries.

REFERENCES

- [1] Liang Tang, Tao Li, Yexi Jiang, and Zhiyuan Chen. "Dynamic Query Forms for Database Queries". IEEE transactions on knowledge and data engineering vol: pp no: 99 year 2013.
- [2] R. Agrawal, S. Gollapudi, A. Halverson, and S. Jeong. Diversifying search results. In *Proceedings of WSDM*, pages 5–14, Barcelona, Spain, February 2009.
- [3] S. Agrawal, S. Chaudhuri, G. Das, and A. Gionis. Automated ranking of database query results. In *CIDR*, 2003.
- [4] S. Boriah, V. Chandola, and V. Kumar. Similarity measures for categorical data: A comparative evaluation.. In *Proceedings of SIAM International Conference on Data Mining (SDM 2008)*, pages 243–254, Atlanta, Georgia, USA, April 2008.
- [5] C. C. Aggarwal, J. Han, J. Wang, and P. S. Yu. A framework for clustering evolving data streams. In *Proceedings of VLDB*, pages 81–92, Berlin, Germany, September 2003.