A Dynamic Query Form System for Mongodb

Anju Abraham
(MTech Computer Science & Engineering, Mar Baselios College of Engineering & Technology, India)

ABSTRACT : *Query forms are a widely used interface for querying the database. When query forms are used, it makes easier for users without much idea about the database schema to access it. Some existing database systems allows user to create customized queries. But creation of such forms is mainly based on the user's familiarity about the schema. So in this paper, a Dynamic query form system an interface capable of creating query forms dynamically to users in NoSQL database like mongodb is proposed. This system starts with a basic query form which will have some important attributes of the database. This form will be filled by the user. When users are interacting with the system their interests will be captured and the query form will be adapted accordingly until the user gets contented with the outputs .The system would then execute the query and the output will be shown. The user is able to respond back to the system about the query results.*

Keywords *–query form, user interest*

1. Introduction

Dynamic query form is an interface which can be used to dynamically create query forms. A database would be as useful as its query interface allows it to be. Users perform many rounds of actions to refine the conditions of the query before reaching the final results. Here the interests of the user during user interactions would be captured and the query form would get adapted in iterative fashion. There will be two types of user interactions. In this system there would be initially a basic query form which will have very few main attributes. This would be improved iteratively through interactions between the user and the system until user becomes contented with results. Mongodb is an open source document database and the leading NoSQL database. These are written as JSON style documents. Mongodb stores all documents as collections. All documents in a collection have a similar related purpose. Document oriented databases are quite different from traditional relational database. If there is a need to add a new column to table, one should change the definition of table itself. By using mongodb the attribute can be added to individual documents without change of other documents.

2. Existing system

The databases of today have a huge and mixed collection of data in them. These contain a number of relations and attributes also. Earlier the developers or the database administrators created the traditional query forms. But now, with the increase in amount of data, it has become difficult to create such static query forms to satisfy the queries. Some existing database systems also allows user to create customized queries. But creation of such queries depends on the users editing manually. The large number of attributes present would confuse the user unless user is not familiar with it.

3. Proposed system

In this paper, a dynamic query form system has been implemented for mongodb, a non-relational database management system enabling users to dynamically generate query forms for users. The users would try to refine their query conditions many times before getting the required results. The system mainly consists of four modules: Metadata processor module, Component ranking module, Quality metric module, Query processor module. Metadata processor module helps to provide information about the database and collections. In the component processor module the selection and projection components would be identified and ranked. In the quality metric module measures like precision and recall would be used to find out the performance of query form. In the query processor module, the system would be executed and result would be obtained.

4. Related work

Some of the databases that exist help users to make query forms. These tools were made by people who very much know the databases. But an end user, who does not know the database, will find it difficult to create such customized queries. Another approach is that the system allows to create automatic static form [1].Here even if a number of such forms are created there is a chance that there can be queries which cannot be satisfied. In another work, a method to generate forms in an automatic way with a database and a sample workload is presented [2].How the quality of forms can be improved is studied in another work. USHER is a system for designing forms, entry and for improving data quality [3]. Studies have been done to improve the query form so that the system would support to a large extend the queries a user may ask. Creating query forms which can satisfy the user demands is an important and difficult task.

5. The basic method of operation

In this system, the interests of the user would be captured during interactions with the user. Initially the user would fill the query form and the system will work and the output would be displayed. If user is satisfied with results, then the system would end otherwise the system continues. There will be two types of interactions: Query form enrichment and query execution [4]. During the first phase, the user would fill the query form and submit it. The system would execute it and the output result would be displayed. During the next phase the user will be able to see a list of query components which are ranked. The user would then be able to select the components into query form currently used. The user can respond back to the system if not satisfied with results. The Fig 1 below shows the basic method of operation of the system.

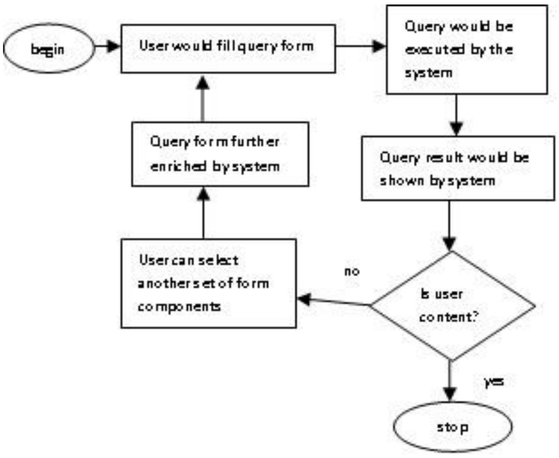

begin
User would fill query form
Query would be executed by the system
Query result would be shown by system
Is user content?
no
yes
stop
User can select another set of form components
Query form further enriched by system

Fig 1 Basic method of operation

6. Metric for ranking

The output required by the user would be given by the query form. Using precision and recall the quality of query output are also found out and measured. On giving different inputs different queries would be created and this would give different output results. To measure the expected performance of form, expected precision and expected recall are used. The proportion of query output which is of interest by current user is expected precision. The expected proportion of data instances which is of interest by user returned by current query form is expected recall [5].

The idea behind ranking is that when users click on a particular query output displayed that data would be of interest to him. The components which user shows interest will be having higher ranks than other components. Based on the expected precision and recall metric, the overall performance metric F-measure would be calculated. The components present in the query form would be ranked in descending order based on Fscore.

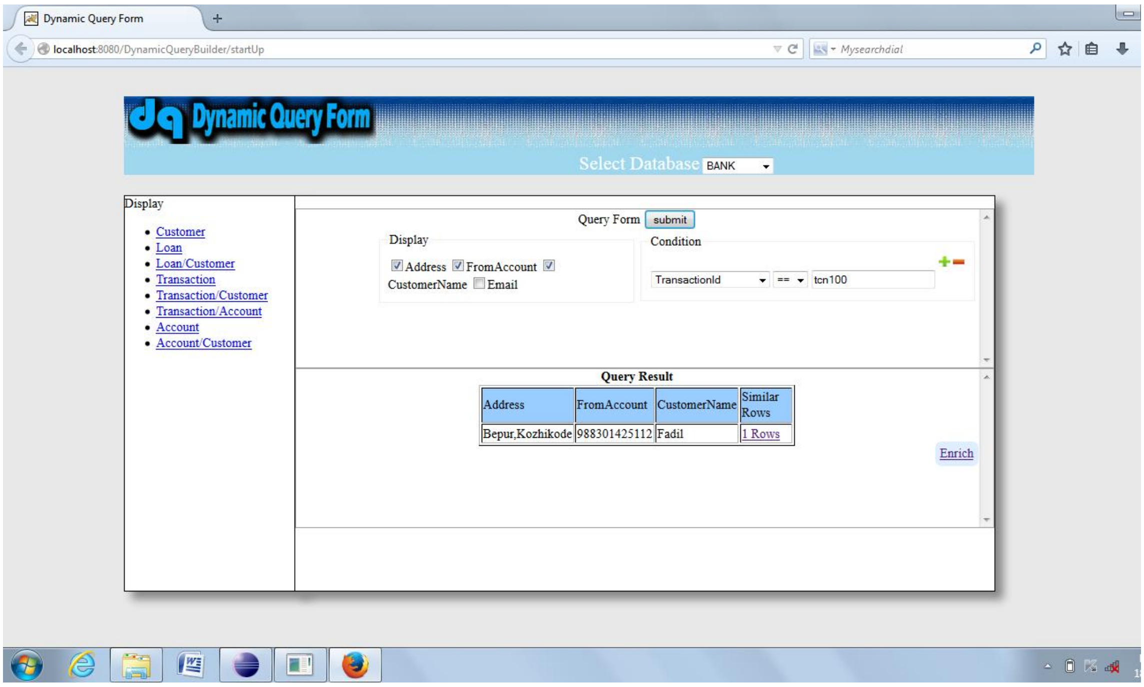
7. Ranking score estimation

In the system, projection components as well as selection components would be ranked. For ranking projection components, entities and also attributes should be ranked. Here the attributes with maximum Fscore would be found out. The score of the entity would be the average Fscore of attributes of the entity[4]. For ranking the selection form components, the attributes which are important should first be found out and then ranked. Based on this query would be constructed and this would be executed. Here the Fscore would be computed incrementally on required attributes. The selection components of query form are relational operators, so from here the best selection components are found out.

Query is actually generated with set of attributes of projection with selection expression. The system provides facility to provide feedback to the system and further enrich the system. During enrichment of query, the query form would be submitted with the data and the query would be made .This would be executed by the database and a resultset would be formed and this would be displayed to user. The user would be able to fill form further and the system would execute this. This process would continue until user is contented with output.

8. Outputs and discussions

In the initial interface of the system, the user can select a database .The system would then display the collections of the selected database in the system. The user can select a particular collection that is of interest to him and now the corresponding display components and conditions can be given. Based on the above, the query would be executed by the system and the query result can be obtained. The user can add more conditions to the query also and get the outputs. The collections which are related to each other would be grouped and based on them the display components and conditions can be given. The Fig. 2 shows the interface of dynamic query form system .In this the various collections of the database bank are shown .The query result displayed is obtained by choosing the display components like address, fromaccount, customer name with condition transaction id= tcn100.The user can further enrich the query form by clicking the enrich button .The system would then display another set of components and from those the desired output can be obtained.



Select Database BANK

Display

- Customer
- Loan
- Loan/Customer
- Transaction
- Transaction/Customer
- Transaction/Account
- Account
- Account/Customer

Query Form

Display

☑ Address ☑ FromAccount ☑ CustomerName □ Email

Condition

TransactionId == tcn100

Query Result

Address	FromAccount	CustomerName	Similar Rows
Bepur,Kozhikode	988301425112	Fadil	1 Rows

Fig.2 Screenshot of query result obtained

9. Conclusion

A form is a way of visually representing a group of operations of query in a way that would be meaningful and efficient. The usage of forms helps users to give conditions on values or choose which values are to be displayed. The options specified by the user would now be executed by the database. The users need not know about how the data is organized in storage or be an expert in query languages. A dynamic query form creation system for mongodb database has been implemented. Through this system user would be able to create query forms and get his required output. The system is able to give output using the individual collections present and also with collections having some relationship with each other in a particular database he chooses.

References

[1] M. Jayapandian and H. V. Jagadish. Automated creation of a forms-based database query interface. In *Proceedings of theVLDB Endowment*, pages 695–709, August 2008.

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

[3] 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.

[4] Liang Tang, Tao Li, Yexi Jiang, and Zhiyuan Chen. Dynamic Query Forms for Database Queries..*IEEE Transactions* on *Knowledge and Data Engineering*, April 2013.

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