Dynamic Query Forms for Database Queries

INTRODUCTION

Modern scientific databases and web databases maintain large and heterogeneous data. Traditional predefined query forms are not able to satisfy various ad-hoc queries from users on those databases.

This paper proposes DQF, a novel database query form interface, which is able to dynamically generate query forms. It helps in querying the Database even for non SQL programmers. We can write general queries and it will be automatically converted into SQL query.

CONCEPTS

Query forms are designed to return the user's desired result. There are 2 traditional measures to evaluate the quality of the query results, precision and recall.

Expected precisions the expected proportion of the query results which are interested by the current user. Expected recall is the expected proportion of user interested data instances which are returned by the current query form.

FScore E(Fi+1) is the estimated goodness of the next query form Fi+1. Since we aim to maximize the goodness of the next query form, the form components are ranked in descending order of FScore E(Fi+1). In the next section, we will discuss how to compute the FScore E(Fi+1) for a specific form component.

The F-score or F-measure is a measure of a test's accuracy. It is calculated from the precision and recall of the test, where the precision is the number of true positive results divided by the number of all positive results, including those not identified correctly, and the recall is the number of true positive results divided by the number of all samples that should have been identified as positive. This score helps in maintaining the relevance of a particular search within Database.

Definition 3. *Given a set of projection attributes A and an universe of selection expressions σ, the* expected F-Measure *of a query form $F=(A_F, \mathcal{R}_F, \sigma_F, \bowtie(\mathcal{R}_F))$ is $FScore_E(F)$, i.e.,*

$$FScore_E(F) = \frac{(1+\beta^2) \cdot Precision_E(F) \cdot Recall_E(F)}{\beta^2 \cdot Precision_E(F) + Recall_E(F)}.$$

APPROACH AND IMPLEMENTATION

WORKING STEPS

1. First step is to setup a method from user to enter query forms in English language and parsing that to our server.

2. We hosted this projects's database on the phpMyAdmin Apache server, then we
have created Node.js localhost server which acts as the user side.

3. All the SQL queries are handled using NPM package named mySQL, using the package we have connected the phpMyAdmin server to the client side and then we can execute queries from client side which gets transferred to backend server and are then executed.

4. The implementation includes the User filling in the query, then the query is parsed and the statement is divided into simpler words each containing some information about different columns from different tables, we created a function named queryBuilder() which recombines these words in such a way that it creates a valid SQL query from the generalized English query written by non SQL user.

5. After the display of the query User has an option to go with the query or re-select queries generated iteratively by the underlying Feed-Forward Neural Network, whenever a query that interests user is generated the User selects the query and the Web App understands what queries need to be generated when fed with specific keywords.

Building the queries using Natural Join

- To get the non SQL queries into SQL queries, we used the technique of One-Hot Encoding, in which we first constructed a list of all possible keywords of interests that user can enter like products, prices, due Payments etc and added them in a dictionary.
- Each time a user enters a query, those keywords from the dictionary are turned Hot (symbolised as 1) and others are remaining Cold (symbolised as 0) which are not in the query as entered by user.
- Based upon the keywords that are turned Hot (or 1) and using the concept of Natural Join, an intermediate dataset is created in JSON format containing the most likely information selected from tables. This information is then sent back to user side from the Express Server as JSON data and can be easily viewed and copied from there.

RESULTS

- To verify the results of the Web Application we wrote different queries related to the product and product lines tables expecting results from that table.
- The app understands the most important Attributes and automatically selects the corresponding Product information from different tables.

As we can see from the database snippet above, we can enter queries in SQL form, semi SQL form, or even general English sentence, we will always get the information from the Product and Product Lines table.

The Web Application creates Natural Join by the important information extracts the relevant data from the table and show it in JSON format.

We have tested the working on all the tables by entering complicated English statements but the App detects the relevant and important information automatically and projects the corresponding results.

productCode	productName	productLine	productScale	productVendor	productDescription
S10_1678	1969 Harley Davidson Ultimate Chopper	Motorcycles	1:10	Min Lin Diecast	This replica features working kickstand, front sus...
S10_1949	1952 Alpine Renault 1300	Classic Cars	1:10	Classic Metal Creations	Turnable front wheels; steering function; detailed...
S10_2016	1996 Moto Guzzi 1100i	Motorcycles	1:10	Highway 66 Mini Classics	Official Moto Guzzi logos and insignias, saddle ba...
S10_4698	2003 Harley-Davidson Eagle Drag Bike	Motorcycles	1:10	Red Start Diecast	Model features, official Harley Davidson logos and...
S10_4757	1972 Alfa Romeo GTA	Classic Cars	1:10	Motor City Art Classics	Features include: Turnable front wheels; steering ...
S10_4962	1962 LanciaA Delta 16V	Classic Cars	1:10	Second Gear Diecast	Features include: Turnable front wheels; steering ...
S12_1099	1968 Ford Mustang	Classic Cars	1:12	Autoart Studio Design	Hood, doors and trunk all open to reveal highly de...
S12_1108	2001 Ferrari Enzo	Classic Cars	1:12	Second Gear Diecast	Turnable front wheels; steering function; detailed...
S12_1666	1958 Setra Bus	Trucks and Buses	1:12	Welly Diecast Productions	Model features 30 windows, skylights & glare resis...
S12_2823	2002 Suzuki XREO	Motorcycles	1:12	Unimax Art Galleries	Official logos and insignias, saddle bags located ...
S12_3148	1969 Corvair Monza	Classic Cars	1:18	Welly Diecast Productions	1:18 scale die-cast about 10" long doors open, hoo...
S12_3380	1968 Dodge Charger	Classic Cars	1:12	Welly Diecast Productions	1:12 scale model of a 1968 Dodge Charger. Hood, do...
S12_3891	1969 Ford Falcon	Classic Cars	1:12	Second Gear Diecast	Turnable front wheels; steering function; detailed...
S12_3990	1970 Plymouth Hemi Cuda	Classic Cars	1:12	Studio M Art Models	Very detailed 1970 Plymouth Cuda model in 1:12 sca...
S12_4473	1957 Chevy Pickup	Trucks and Buses	1:12	Exoto Designs	1:12 scale die-cast about 20" long Hood opens, Rub...