A REVIEW ON REVERSE APPROACH FOR SQL QUESTION GENERATION

Miss. Abhilasha A. Dwivedi[1], Dr. Dinesh D. Patil[2]

[1] *M. Tech Student, Department of Computer Science and Engineering, Shri Sant Gadge Baba College of Engineering & Technology, Bhusawal - 425203, Maharashtra, India.*
[2]*Associate professor & Head, Department of Computer Science and Engineering, Shri Sant Gadge Baba College of Engineering & Technology, Bhusawal - 425203, Maharashtra, India.*

ABSTRACT

Structured Query Language (SQL) is a language which inserts, manipulates and retrieves data stored in a relational database. It is an essential knowledge that all computer science students must have. With the aim of making students learn SQL appropriately, the instructor spends a lot of time and effort to prepare SQL exercises for teaching SQL queries to them. However, with time limitation, the SQL instructor or teacher had to reuse the old exercises which have small number of exercises with a lesser amount of diversity in SQL commands. Due to this, the students might not have sufficient exercises to meet their need. And also practicing with the same exercises repeatedly cannot help in improving student's learning outcome. Thus, instructors or teachers and students or SQL learners, all require a way which can benefits both of them. This paper reviews the ways which uses the reverse approach that is creating the answers first and then the questions are generated from the answers. This methods not only saves the time of instructors in creating exercises but also provides the students or learners with the variety of questions for practice.

Keywords- *reverse question generation, SQL, automatic question generator, e-learning*

1. INTRODUCTION

Information technology has become one of the most substantial parts of our lives, which can be either any software, system, device such as mobile, computer, etc. websites, or others and now in 21st century we cannot even imagine life without it. The most important part that is used to drive these technologies is the data which is stored in the database. For example, in business domain, manager keeps track of stock, customers, purchase, sales, and customer receipts and so on, to improve their inventory management and also it helps with decision making, such as how much and when to order new stocks. Similarly, a social media website stores its user's data such as profiles, posts and photos, etc. in the database and connects to the programming language, like PHP, Java Server Pages, etc., to display the information on the page. Consequently, the knowledge in the field of database, like relational database management system (RDBMS) and structured query language (SQL) has become very important for a career mainly related to technology, such as a programmer, data analyst, database manager, etc.

Structured query language (SQL) is a language intended for managing data. Basic SQL statements such as SELECT, FROM and WHERE clauses are a key basis when studying or learning practical database courses either in college or from any other institute. In the same class, there might be some students who beforehand know some SQL queries and who might want to learn more new ones. Also there might be students who do not know a single SQL query. If the questions in the exercises are static and students practice repeatedly with the same set of exercises, then their knowledge will not improve because they may answer the questions from memory, without understanding the queries properly. Also, students may become bored because there is no advancement in complexity of the exercises. In teaching and learning management systems, every instructor or teacher must know how their students should learn in both theoretical and practical extents. Usually, the instructor sets the learning targets for testing each SQL query and writes SQL questions based on these learning targets. As a result, the process of creating exercises becomes time consuming.

There are mainly two methods to prepare SQL exercises for teaching SQL. The first method is to prepare the exercises by taking database schema and here the benefit is for instructors because they do not have to validate the answer which is returned. On the other hand, students have to imagine their results by themselves. The second method is to create exercises with reference to both schema and data set which makes students learn and try the query with real database which results in more effective learning. However this method creates problems for instructors since it consumes more time and effort. These exercises which are created from schema and dataset involves SQL question, SQL query and answer validation for which instructor spends a lot of time and effort. But due to limitation of time, the instructor reuses the old exercises which might have lesser quantity of exercises having little varieties of questions. Existing systems have also some restrictions on the amount of supported SQL statements [11].

However using reverse approach, can have number of advantages such as using it in any system makes students to practice their SQL skill with diversity exercises and the existing question can be avoided, teachers or instructors can reduce their workload to create SQL exercises and can save time and effort, it improves opportunity for SQL skill enhancement for students, meet the learning purpose also then the learning result from student is more efficient, etc.

2. LITERATURE REVIEW

2.1 Automatic question generation

In order to teach and test the students or learners, instructor or educator conducts the examinations or provides the students or learners with the practice tests and exercises, however for this it is essential for them to generate the questions manually which is a time consuming procedure. Thus researchers have decided to develop the methods, systems or algorithms by which, automatic questions can be generated, with the help of which the required time and efforts can be reduced.

Developing Automatic Question Generation systems became one of the important research issues because it requires understandings from a range of disciplines, such as Artificial Intelligence, Natural Language Understanding, and Natural Language Generation. There are two types of question formats, multiple choice questions which requests a word in a given sentence, the word may be an adjective, adverb, noun, vocabulary, etc., the second format is the Text to Text Question Generation that requests a word or phrase corresponding to a particular entity in a given sentence. In order to develop an automatic question generator, a number of researchers have presented their effort and many algorithms or methods are proposed to develop the automatic question from specified or given text.

To automatically generate question by creating answer first is a better approach to lighten the instructor's workload and enhance learning practices for students. By reviewing the reverse approach for question generation, basically four approaches are found to be used.

1. “questions by mutation” mutates the specified text and generates questions from it.

Lee *et al.* in “Generating Grammar Questions Using Corpus Data in L2 Learning” has used collection of errors which were found by analysing existing corpora and then generated questions having common errors which the students have to identify [2] and explored automatic generation of English grammar questions based on statistical machine learning techniques. Generating grammar question reviewed in this paper works in two ways: 1) It can realistically determine error-prone positions when generating questions by using L2 learner corpora, and 2) Recognition of errors is performed on the basis of real students’ grammatical errors by means of using several context features such as bigram, trigram, and dependency structures.

In the same way, Funabaki *et al.* in “Toward Personalized Learning in JPLAS: Generating and Scoring Functions for Debugging Questions” has created debugging questions for programming courses [3], [4]. Algorithm in this paper generates bugs in JAVA source code by using one or more of three approaches which are command deletion, variable swapping and command insertion. It basically implements the automatic generating function and the scoring function concerning debugging questions in JPLAS (Java Programming Learning Assistant System).

2. “questions by keyword” use keywords from the input and then creates a new question from that extracted keyword.

Liu *et al.* in “Using Wikipedia and conceptual graph structures to generate questions for academic writing support” presents an intelligent question generator tool that supports writing literature reviews. It helped students improve their literature reviews by parsing sentences from their reviews, extracting key phrases and

finding appropriate sentences from Wikipedia pages, then forming questions about those sentences [5]. These questions help students in improving their reviews without requiring comebacks of the instructor.

3. "questions from input", generates the questions from the input such as images.

Jain *et al.* in "Creativity: Generating diverse questions using variational auto encoders" generated questions from the input image [6]. Their research algorithm can be applied in a number of fields, e.g. educational study, driving assistance, entertainment, etc. They suggested a creative algorithm for generating visual questions which associates the benefits of variational autoencoders with short-term memory networks and also indicate that their proposed framework is capable to generate a large set of variety of questions for a single input image.

4. "questions from metadata", generates the questions from the metadata.

Abdul Khalek and Khurshid's "Automated SQL query generation for systematic testing of database engines" generates new SQL queries from the database schema for testing database performance [7] which is capable of automatically generating syntactically and semantically appropriate SQL queries for testing, and input data to populate test databases, and also expected result of executing the given query on that generated data. Its result shows that one can not only automatically generate valid queries which detect bugs in database engines, but also perform automated testing of database engines. However, it did not consider data in tables or generate a text explanation of the query which was to be used as an SQL question in exercises.

All of the above research papers fundamentally converts a source constituent into a new constituent i.e. question. This source constituent can be either the solution of the question, such as a grammatically correct sentence or bug free source code, or something more complex, such as an input image or literature review of a student.

2.2 Reverse SQL Question Generation (RSQLG) Algorithm

Thanakrit Julavanich, Srinual Nalintippayawong and Kanokwan Atchariyachanvanich, in "RSQLG: The Reverse SQL Question Generation Algorithm" proposed an algorithm that uses the reverse method by generating the SQL queries first instead of SQL questions [1]. RSQLG algorithm reverses the manual question writing process by starting with the answer. This algorithm used a database as input to generate the SQL query and, from the query, generated the text description as a question.

The RSQLG reverses the traditional process by creating an SQL query (answer) first. The query from first step is an input to generate query explanation (question) and is validated simultaneously. The RSQLG studies the existing data and database structure by using various constraints. The instructor or teacher can also specify the format of exercise, its language and explanation of the questions accordingly. The RSQLG algorithm supports only DML commands such as SELECT, INSERT, UPDATE and DELETE.

Working of RSQLG

Fig-1 represents flow of RSQLG Algorithm.

Database Pre-Processing-Database pre-processing extracts the data and metadata necessary for SQL query creation, e.g. database schema, table schema, attribute data type, attribute data length, and attribute constraints, relationship and key. Due to this, RSQLG will not need to connect to the real database, whenever it needs some data and consequently it reduces the time to generate exercises.

Query Metadata Generator-Query Metadata Generator generates the SQL queries and associated text explanations. The Query Metadata Generator algorithm works on simple SQL clauses such as FROM, WHERE and SELECT. It generates the metadata of output query. And then the SQL answer generator and SQL question generator use this metadata for creating queries and questions respectively. Thus the metadata is converted into SQL query (answer) and query explanation (question).

The SELECT command generation triggers all modules in the complete flow and the RSQLG algorithm chooses related tables and creates conditions to query the result.

The INSERT command generation contains FROM, WHERE, SELECT modules. The algorithm creates the new records which concatenates the existing data with the same column formats or do some changes, e.g. adding prefix, suffix and deleting some words. The new records are result from Insert command which come from student's answer.

The UPDATE command generation includes the same modules as the INSERT command. The algorithm retrieves one or more records which will be modified by student or learner. The process uses same approach as WHERE clause in SELECT command generation then generate new data in one or more column by using the same approach as INSERT command generation.

The DELETE command generation contains FROM, WHERE and LIMIT modules. The algorithm creates a condition which performs to delete records with the same method as WHERE clause in SELECT command generation.

Query Metadata Bank-Query metadata bank will store the generated metadata in modular objects instead of storing the whole output which can result in reducing the size of output. The first object is FROM object which will store the table and will join table data. The second object is SELECT object which will store the selectable columns from tables and new values generated from INSERT, UPDATE and DELETE query. The final object is WHERE which will store the valid conditions for queries and questions.

SQL Answer Generator-After the generation of query metadata, the metadata from metadata bank will be processed to generate the SQL query. In order to generate an exercise, the SQL answer generator retrieves the data from query metadata bank which will be then associated to the defined objective in question setting. The data from Query metadata bank is joined as a complete query metadata for query generation.

SQL Question Generator-The SQL question generator expands the query metadata output from the generator with a text explanation. This is a natural language description of the query by splitting out each part of query and replacing it with natural language. Table-1 represents the example of the natural language description of the associated query and how the SQL question generated from the query looks.

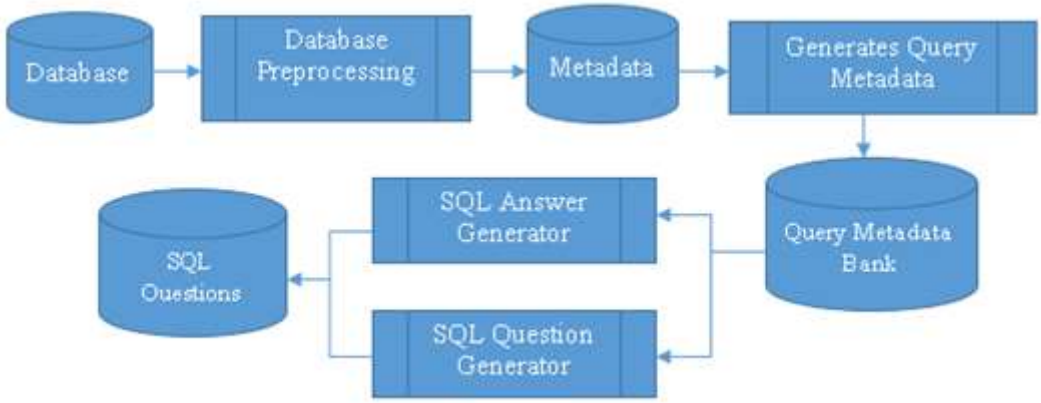

Fig-1 RSQLG flow

Table-1 Example of output question

Separated parts	SELECT	*	FROM	EMPLOYEES
Explanation	Display	information	Of	All employees
SQL Question	Display information of all employees			

3. BACKGROUND

3.1 Structured query language (SQL)

SQL is used to communicate with a database. According to ANSI (American National Standards Institute), it is the standard language for relational database management systems mainly envisioned for managing the data. SQL statements are used to perform tasks such as update data on a database, or retrieve data from a database [12].

SQL is very popular language which has several features as follows: It allows users to access data in the relational database management systems, to describe the data, to define and manipulate the data in a database, to embed within other languages using SQL modules, libraries & pre-compilers, to create and drop databases and tables, to create view and stored procedure, to create the functions in a database, to set permissions on tables and procedures and views, etc.

SQL Commands:

There are certain standard SQL commands which are widely used to interact with relational databases such as CREATE, SELECT, INSERT, UPDATE, DELETE, etc. These commands are popularly classified into the following groups based on their nature:

1. Data Definition Language (DDL)
2. Data Manipulation Language (DML)

Table-2 and Table-3 represents the fundamental Data Definition Language (DDL) commands and Data Manipulation Language (DML) commands respectively.

Table-2 Data Definition Language commands

Command	Description
CREATE	Creates or generates a new table in the database.
ALTER	Alters or modifies an existing table in the database.
DROP	Deletes an entire table of the database.

Table-3 Data Manipulation Language commands

Command	Description
SELECT	Retrieves records
INSERT	Creates a record
UPDATE	Modify records
DELETE	Delete records

3.2 Existing Systems

The traditional method of creating SQL exercises begins from the database analysis and then a query explanation (question) is created. The instructor writes the SQL query for SQL explanation and validates result with the real database which represented in Fig-2. That means, earlier, the instructor usually begins with exploring the database schema and the data itself. Then, a question is modeled based on a query that a student should understand and associated with a learning objective. Next, the instructor specifies the correct SQL query to answer the question. Lastly, this query is then validated using the real database to determine whether the result is returned as specified in the question or not [8]. Also, in some systems queries are generated automatically [9]. Also validation of the queries might be a challenging task since it includes occupying a test database which may contain several tables and then examining the results of the query execution on the test database [10].

Disadvantages:

- It requires huge amount of time and effort to create exercises
- Students get repeated questions
- Number of questions are very less to satisfy student's needs.
- Creating variety of exercises with diverse questions requires large amount of time and efforts which can increase the instructor's workload.

Fig-2 Generating SQL exercises manually

3.3 Reverse Approach

The reverse approach in generation of questions and answers can reverse the manual question creation process which starts creating query answer first instead of creating questions first. This approach has ability to generate bulk questions with less effort. It is a different approach to lighten the instructor's workload and enhance practices for students. It uses a database as input to generate the SQL query and, from the query, generated the text description as a question [1]. Fig-3 represents the input and output of Reverse Approach. Table-4 represents the basic comparison between the traditional approach being used by most of the existing systems and reverse approach.

Fig-3 Generating SQL exercises using reverse approach

Table-4 Comparison between traditional method and reverse approach

Traditional Method	Reverse Approach
Creates questions first	Creates answers first
Generates questions manually	Generates questions automatically
Generates less variety of questions	Generates diverse questions
Increases workload of instructor in creating exercises	Decreases workload of instructor in creating exercises
Requires great effort and time	Requires less effort and time
Improves SQL skills	Improves SQL skills better than traditional method
Learning is less effective	Learning is more effective than traditional method

4. CONCLUSIONS

The reversed SQL question generation algorithm (RSQLG) used in the reverse approach reverses the manually writing exercise process from creating the question thereby beginning with the generation of answer first which can be used to solve problems in teaching and learning SQL. The RSQLG took a database as an input then it generated query metadata by looking into database schema and data set. The generated query metadata needs to be matched with the defined constraints. The query metadata can be converted to valid SQL query (answer) and explanation of the query in regular natural language (question). The instructors can set the language, format and question explanation by themselves. The students can select the objective of practice without receiving identical or repeated questions thereby improving SQL skills of students. It has several advantages such as it reduces time of instructor in creating exercises, students get varied questions for practicing, improves SQL skills of students, learning results of students are more efficient than traditional approach, requires less effort and time, decreases workload of instructor of creating questions manually, which saves their time and so on.

The RSQLG requires some improvement in complex exercises, language support which has more complex grammar and more unsupported commands, e.g. DDL commands. It can also improve the text explanation with the aim of supporting other languages having more complex grammar. RSQLG Algorithm can be combined with some other methods or can be modified so that it can be used in any type of database e-learning system, in colleges to teach students or in computer institutes for practicing SQL.

5. ACKNOWLEDGEMENT

I feel great pleasure in submitting this paper on "A Review on Reverse Approach for SQL Question Generation". I wish to express true sense of gratitude towards my Principal Dr. R. P. Singh and special thanks to my guide and H.O.D., Dr. Dinesh D. Patil who at every discrete step in preparation of this paper contributed his valuable guidance and help to solve every problem that arose. Also I thank all the researchers and authors of reference papers referred by me for this paper which is a review of various researches done in the field of automatic question generation. Also, most likely I would like to express my sincere gratitude towards my parents for always being there and for their continuous support and encouragement.

6. REFERENCES

[1] Thanakrit Julavanich, Srinual Nalintippayawong and Kanokwan Atchariyachanvanich, "RSQLG: The Reverse SQL Question Generation Algorithm," *in IEEE 6th International Conference on Industrial Engineering and Applications, 2019.*

[2] K. Lee, S.-O. Kweon, H. Seo, and G. G. Lee, "Generating grammar questions using corpus data in L2 learning," in *Proc. IEEE Spoken Lang.Technol. Workshop (SLT)*, Dec. 2012, pp. 443_448.

[3] N. Funabiki, T. Mohri, and S. Yamaguchi, "Toward personalized learning in JPLAS: Generating and scoring functions for debugging questions," in *Proc. IEEE 5th Global Conf. Consum. Electron.*, Oct. 2016 pp. 1_4.

[4] S. Yamaguchi, T. Mohri, and N. Funabiki, "A function for generating debugging questions in a Java programming learning assistant system," in *Proc. IEEE 4th Global Conf. Consum. Electron. (GCCE)*, Oct. 2015, pp. 350_353.

[5] M. Liu, R. A. Calvo, A. Aditomo, and L. A. Pizzato, "Using Wikipedia and conceptual graph structures to generate questions for academic writing support," *IEEE Trans. Learn. Technol.*, vol. 5, no. 3, pp. 251_263, Jul. /Sep. 2012.

[6] U. Jain, Z. Zhang, and A. Schwing, "Creativity: Generating diverse questions using variational autoencoders," in *Proc. IEEE Conf. Comput. Vis.Pattern Recognit. (CVPR)*, Jul. 2017, pp. 6485_6494.

[7] S. A. Khalek and S. Khurshid, "Automated SQL query generation for systematic testing of database engines," in *Proc. IEEE/ACM Int. Conf. Automated Softw. Eng.*, Sep. 2010, pp. 329_332.

[8] Bikash Chandra, BhupeshChawda, and BiplabKar, "Data Generation for Testing and Grading SQL Queries", may 2017.

[9] Q. Do, R. Agrawal, D. Rao, and V. Gudivada. "Automatic Generation of SQL Queries". In: Proceedings of the 121st ASEE Annual Conference & Exposition. ASEE, June 2014, pp. 1 – 11.

[10] Claudio de la Riva, María José Suárez-Cabal, Javier Tuya, "Constraint-based Test Database Generation for SQL Queries", in Proceedings - International Conference on Software Engineering, May 2010, pp. 266-276.

[11] K. Atchariyachanvanich, S. Nalintippayawong, and T. Permpool, “Development of a MySQL sandbox for processing SQL statements: Case of DML and DDL statements,” in Proc. 14th Int. Joint Conf. Comput. Sci. Softw. Eng. (JCSSE), Jul. 2017, pp. 1–6.
[12] Fernando Almeida, "Practical SQL Guide for Relational Databases ", January 2016.