Review of Automated Assessment Tools for grading student SQL queries

Sidhidatri Nayak
Amity Institute of Information Technology
Amity University
NOIDA, India

Reshu Agarwal
Amity Institute of Information Technology
Amity University
NOIDA, India
ragarwal1@amity.edu

Sunil Kumar Khatri
Amity University, Tashkent
Uzbekistan
skkhatri@amity.edu

2022 International Conference on Computer Communication and Informatics (ICCCI) | 978-1-6654-8035-2/22/$31.00 ©2022 IEEE | DOI: 10.1109/ICCCI54379.2022.9740799

***Abstract*— The Structured Query Language (SQL) is the most widely used language for querying relational databases and it is taught in all basic programming classes in higher education. Even though SQL is the most basic, simple, and most well-structured language, it is hard to learn due to traditional approaches adopted by learners. In recent years, several online tools have been developed to learn and assess the SQL query. The automated grading of SQL queries will also be useful for the instructors who compile grades manually. In this paper, we have discussed different types of errors students may encounter while writing SQL queries and how correct evaluation and timely feedback can help students to improve their SQL knowledge. In this paper, we also have reviewed and discussed the strength and weaknesses of the existing semi and fully automated assessment tools evolved over several years for the automatic assessment of SQL queries submitted by students and the need for a new fully automated system for the assessment of student SQL queries which can overcome the weaknesses of the existing tools.**

Keywords- SQL; Automated Assessment Tools; Query Formulation Tools;

I. INTRODUCTION

The introductory database courses are the most elemental part of the software development/computer engineering courses at higher education. The objective of the databases course is to learn how to create a physical database, how to conceptually design the database, and how the user views the data from the database [1]. The Structured Query Language (SQL) is the language that construct, manipulate, retrieve and also manage data in the database. It is an open-source language and easier to learn than any other existing programming [2]. It is essential for the students at introductory programming courses to learn good relational database concepts including how to organize the data, removing inconsistency and redundancy. This can be applied to most application development that is focused on data-driven applications. So, SQL is an essential and valuable programming language students should acquire and be taught at the university level [3]. Assessment and feedback are two important parameters of the learning [4]. Assessment helps in the improving the query writing skill in student [5]. The SQL instructors require Automated Assessment Tools (AAT) to teach query construction techniques and maintain uniformity in grading and reduce the manual load [6].

II. ASSESSMENT OF SQL QUERIES

A. Error categorization in SQL

SQL query gives the correct output if the query is constructed logically and written with proper syntax, otherwise it produces syntax errors or incorrect output. The basic operations students performed are defining the data, manipulating the data, controlling the data and transaction. SQL queries we write are CREATE, INSERT, SELECT, UPDATE, and DELETE. The SELECT statement is used to retrieve the data which includes restriction condition through WHERE, grouping of rows through the GROUP BY clause, restrict the group result by HAVING, and sort the records by the ORDER BY clauses. The SQL queries can be written in seven different types including group by, having, different types of joins etc. [7,8].

Students write SQL queries for the assignment given by the instructor as part of the learning process. Then the student's solutions are thoroughly evaluated, and grading is done. The students are provided with feedback to improve their overall learning approach. The SQL statement written by students should be assessed properly to find their errors and categorize their errors. The common SQL errors are categorized as synonyms errors, syntax errors, and semantic errors [18,19].

The Synonyms Errors in SQL occur in students' SQL queries because errors such as replacing where with if as they sound similar [20,21]. The syntactic error we may encounter during query execution are, undefined column, Grouping error, undefined table, Ambiguous column, Undefined function, Data type mismatch, invalid date-time format, Invalid schema name, Undefined object, Duplicate alias, Invalid column reference, ambiguous function, etc. [22,23]. But semantic error comes when student's do not get logically correct results, but the query is syntactically correct. The different error categories were defined based on the above three types of errors.

B. Assessment SQL Queries

Manual assessment of a student's SQL statement is time-consuming and a difficult task to assess the logical errors and check if the query returns the correct set of results. Proper immediate feedback is not possible in manual assessment

978-1-6654-8035-2/22/$31.00 ©2022 IEEE

process. The correct query formulation skill involves conceptualizing, constructing the query, executing them, visualizing the result, editing the query till it produces the correct result. The student learning approach id directly proportional to correct grading system, and hence should be carefully designed and implemented. Hence an Automated Assessment Tool is required to assess the SQL queries written by students, generating feedback and grades them.

The objective of this paper is to review the common syntactic and semantic mistakes in student's SQL answers. This paper is going to recite the flaws with the existing SQL learning and grading systems [24]. This paper states the cons of manual marking system, as well as reviewing the current Automated Assessment tools for SQL and analyzing how they work and what methodologies they have followed to evaluate SQL statements and followed by a comparative analysis. This paper outlines the need for creating a new paradigm for automatically assessing SQL statements and providing rapid feedback. [25].

III. Related Work

In this paper, we have discussed various automated assessment tools for learning, formulation of SQL queries and automatic grading of student's submissions and generating feedback. This review outlines the automated assessment tools used for assessing the SQL statement submitted by the student and improves the overall learning of the SQL in introductory programming courses. Kleiner, Tebbe, Heine [7] have discussed the reviews of students after using automated assessment tools in SQL learning and the assessment process. Some students find it beneficial while some find it difficult based on certain parameters. The negative and positive aspects of using automated assessment systems are discussed in figure: 1.

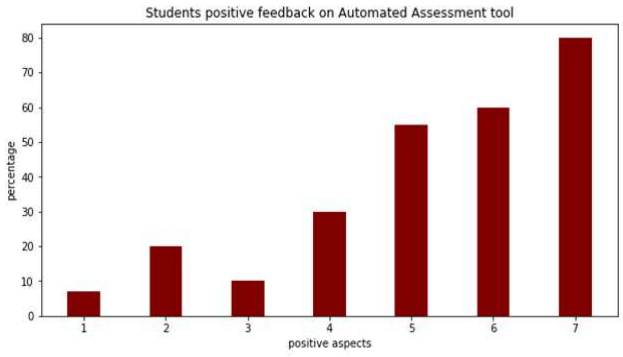

Figure 1: Positive aspect of Automated Assessment tool

TABLE I. POSITIVE ASPECTS

Positive Aspects	Description
1	Not Helpful
2	Time Management
3	Remote Submission
4	Creative Thinking
5	No Personal Instruction
6	Correct Grading
7	Immidiate Feedback

Table 1 describes the positive aspects considered in the above figure.

A. Automated Assessment Tools

The Automated Assessment Tools for SQL focus on the student's common syntactic and semantic mistakes in the SQL queries they have submitted. We are going to discuss some of the assessment tools which uses both summative and formative assessment.

1. SQL Tester [2] is an automated assessment tool for grading SQL queries, which is based on simpler questions. It does not include complex queries like correlated subqueries or self-joins and complex group by queries. The test time is generally 50 minutes which can be extended based on the student's level. This uses simple schemas and simple ER Diagrams. SQL Tester matches the student's submission with the desired output and marks it as correct if it matches the model output. The student's statement must be identical to the model answer, for example, in the same order and in the same case.

2. SQL-Statement Grader automatically assesses SQL statements in any introductory database course [7]. This grades the student's solution to different questions. Students can resubmit their solutions for improvement. A report is generated after each evaluation in XML format.

3. SQL Knowledge Tester [27,28] uses new question generation and answers assessment. It formulates questions on a sample database, the student write queries and the Knowledge Tester evaluates the queries and generates feedback. It compares the similarity between student solution and the model solution. The student's submission is graded as correct only it is 100 % like the model solution. Students could debug previous solutions if they failed the question formulated by the knowledge tester. The SQL-Lab helps students to compose as well as run queries, compare the results, and check the how efficient is the student's SQL query.

4. SQLify delivers automatic grading and reduces instructor's marking load [14]. The SQLify system provides semantic feedback and improves query put up skills. Students may update their solutions indefinitely for a better solution and submit their work. The student's work then reviewed by peers and finally the SQLify grades and generates feedback.

5. Active SQL is an integrated learning platform for learning and online assessment and provides feedback after analyzing results of the SQL query [9]. The grading system gives marks in percentage. The percentage ranges from 0-100%. The student's submission is compared with the model output.

6. SQLator is an interactive tool, which analyze the SQL syntax and grade them as correct query or incorrect query [10]. The student attempts multiple trials before the final

submissions. It decreases the marking time, efficient and provides prompt feedback.

7. AsseSQL had seven categories of questions. AsseSQL uses complex schemas. students face difficulties in remembering the database structure and therefore commit errors in writing the relation names and attribute names. Students may undergo multiple attempts.

Table 2 demonstrates the overview of SQL automated assessment developed so far and feedback generation tools developed from 2003 to 2018

TABLE II. COMPARISION OF VARIOUS SQL ASSEMENT TOOLS

Tools	Author	Year	Syntax error	Model Solution	Semantic Error	Grading	Feedback to students
SQL Tester	Kleerekoper, Schofield	2018	Yes	Yes	Yes	Correct/ Incorrect	No Feedback
SQLg	Kleiner,Tebbe , Heine	2013	Yes	Yes	No	Score	No Feedback
SQL-KNOT	Brusilovsky et al	2010	No	Yes	Yes	Correct/ Incorrect	No Feedback
SQLify	Raadt, Dekeyser and Lee,	2006	Yes	No	Yes	Conjunctive	Yes
Active SQL	Cumming and Russell	2005	Yes	Yes	Yes	Percentage	Not proper Feedback
SQlator	Sadiq et al.	2004	No	No	Yes	Correct/ Incorrect	Not proper Feedback
AsseSQL	Prior	2003	No	No	yes	Correct/ Incorrect	No Feedback

IV. COMPARISION

According to prior SQL tool research, SQL-KnoT, AsseSQL, and SQLator all have one thing in common: a student's solution must always produce the same result as the model solution. This feature is part of the grading process and allows students' answers to be labelled as correct or incorrect. Both ActiveSQL and SQLator are unable to validate SQL syntax. Student queries are solely given binary marks by SQLator and AsseSQL, with no comments or suggestions for improvement. They just use the system to create a single communication channel between the learner and the instructor. More professional features, such as better grading and feedback, are included in SQLg, SQLify, and ActiveSQL. SQLg has more functionality and is of higher quality. Additionally, after finishing the grading process, SQLator does a thorough check to verify consistency. Although SQLify and ActiveSQL are similar in certain aspects, SQLify evaluates semantic SQL statements to a higher level before comparing them to the reference model. The systems mentioned above, on the other hand, concentrate on the final responses supplied without offering any detailed feedback on why a response is erroneous. SQL Tester allows students to answer questions in any order they choose and make as many attempts as they want within the time limit; as a result, students frequently make many attempts until they find out what faults they're making. SQL Tester allows you to choose from a variety of SQL question categories, as well as the number of trials and whether to assess students' answers as correct or incorrect. The AsseSQL does not appear to present the schema of the relevant database with questions and model answer output, like the SQL Tester does. One of the difficulties with AsseSQL is that test marking is binary, suggesting that answers are either correct or incorrect. No marks are given if a student's answer is just partially right. As a result of the system's uneven distribution of marks among learners, AsseSQL may have an impact on the evaluation process.

Table 3 summarizes current SQL assessment and learning tools, as well as their drawbacks. According to the data, none of the current systems provide sufficient feedback, which is a vital part of the learning and grading process for both instructors and students.

TABLE.III. LIMITATION OF AUTOMATED ASSESSMENT TOOL

SQL Tester	Match student's submission with the model answer. case sensitivity is not handled. No feedback.
SQLg	Candidates can practice before the assessment. provide more details of manual grading.
SQL-KnoT	Minimized grading system
SQLify	Reviews by given by multiple reviewers. No proper explanation of errors
ActiveSQL	Grading in percentage. No explanation of mistakes. Limited feedback system
SQLator	Grading as correct/incorrect. No exact explanation
AsseSQL	Grading as correct/incorrect. No description of errors

V. CONCLUSION

This review paper states various aspects of Automated Assessment systems to improve SQL learning for students as well as instructors. Different review of existing SQL assessment systems was specified and corelated to discuss the advantages and disadvantages. This paper examined the general methodologies for formative assessment of SQL queries. The agenda of this review paper is to point out current SQL learning and grading solutions have flaws and hence needs to be corrected. Most papers followed the partial assessment of student SQL queries and offer limited feedback and are not personalized. This paper highlights the limitations of each tool and discusses the need for a new assessment tool that can overcome the limitations and help students learn SQL efficiently.

REFERENCES

[1] Lye, S. Y., & Koh, J. H. L. (2014). Review on teaching and learning of computational thinking through programming: What is next for K-12? Computers in Human Behavior, 41, 51–61.

[2] Kleerekoper, A. and Schofield, A. (2018) 'SQL tester: an online SQL assessment tool and its impact', in Proceedings of the 23rd Annual ACM Conference on Innovation and Technology in Computer Science Education - ITiCSE 2018. New York, New York, USA: ACM Press, pp. 87–92.

[3] Michael de Raadt, Stijn Dekeyser, and Tien Yu Lee.(2007). A system employing peer review and enhanced computer assisted assessment of querying skills. ,163–178 pages.

[4] M.Y. Santos, C Costa, J.Galvão(2017) , Evaluating SQL-on-Hadoop for Big Data Warehousing on Not-So-Good Hardware, IDEAS, Proceedings of the 21st International Database Engineering & Applications Symposium, July 2017 PP 242–252.

[5] Armbrust, M. et al.(2015). Spark sql: Relational data processing in spark. Proceedings of the 2015 ACM SIGMOD International Conference on Management of Data , 1383—1394.

[6] Carsten Kleiner, Christopher Tebbe, and Felix Heine.(2013). Automated grading and tutoring of SQL statements to improve student learning. In Proceedings of the 13th Koli Calling International Conference on Computing Education Research,pp. 161–168.

[7] Brown, S. and Knight, P. (1994). Assessing Learners in Higher Education. London: Kogan Page. Pp.161

[8] Cumming, A. and Russell, G. (2005) 'Automatic checking of SQL: Computerised grading.', International Journal of Learning, 12(3), pp. 127–134

[9] Sadiq, S. et al. (2004) SQLator: an online SQL learning workbench, in ACM SIGCSE Bulletin. Leeds, United Kingdom: ACM, pp. 223–227

[10] Prior, J., and Lister, R.(2004). The backwash effect on SQL skills grading. ITiCSE 2004, Leeds, UK. pp. 32-36

[11] Ala-Mutka, K. M. (2005) A survey of automated assessment approaches for programming assignments, Computer Science Education, 15(2), pp. 83–102

[12] Buyrukoglu, S., Batmaz, F. and Lock, R. (2016) Semi-automatic assessment approach to programming code for novice students. Available at: https://dspace.lboro.ac.uk/2134/20477

[13] Stijn Dekeyser, Michael de Raadt, and Tien Yu Lee. 2007. Computer assisted assessment of SQL query skills. In Proceedings of the eighteenth conference on Australasian Database-Volume 63. Australian Computer Society, Inc., 53–62

[14] Alireza Ahadi, Vahid Behbood, Arto Vihavainen, Julia Prior, and Raymond Lister.(2016). Students' Syntactic Mistakes in Writing Seven Different Types of SQL Queries and its Application to Predicting Students' Success. In Proceedings of the 47th ACM Technical Symposium on Computing Science Education - SIGCSE '16. Pp.401–406

[15] Alireza Ahadi, Julia Prior, Vahid Behbood, and Raymond Lister.(2015) A Quantitative Study of the Relative Difficulty for Novices of Writing Seven Different Types of SQL Queries. In Proceedings of the 2015 ACM Conference on In-novation and Technology in Computer Science Education - ITiCSE '15. Pp.201–206

[16] Wetly, C. 1985. Correcting user errors in SQL. International Journal of Man-Machine Studies 22(4): 463-477

[17] Buitendijk, R. B., 1988. Logical errors in database SQL retrieval queries. Computer Science in Economics and Management 1 (1988) 79-96

[18] Mitrovic, A. (1998). Learning SQL with a computerized tutor. SIGCSE '98, pp. 307--311.

[19] Christian Goldberg, Stefan Brass, Semantic Errors in SQL Queries: A Quite Complete List, Fourth International Conference on Quality Software, 2004. QSIC 2004. Proceedings.

[20] Stefan Brass, Christian Goldberg. Detecting Logical Errors in SQL Queries. Technical Report, University of Halle, 2004

[21] Ke, H., Zhang, G. and Yan, H. (2009) Automatic grading system on sql programming, in Scalable Computing and Communications; Eighth International Conference on Embedded Computing, 2009. SCALCOM-EMBEDDEDCOM'09. International Conference on. IEEE, pp. 537–540. doi: 10.1109/EmbeddedCom-ScalCom.2009.pp.105

[22] Kearns, R., Shead, S. and Fekete, A. (1997) A teaching system for SQL, in Proceedings of the 2nd Australasian conference on Computer science education. ACM, pp. 224–231.Page | 184

[23] Buitendijk, R. B.(1988). Logical errors in database SQL retrieval queries. Computer Science in Economics and Management 1 (1988) 79-96 R. Kearns, S. Shead, and A. Fekete. 1997. A teaching system for SQL. In Proceedings of the second Australasian conference on Computer science education - ACSE '97. ACM Press, New York, New York, USA, pp.224–231

[24] Brusilovsky, P. et al. (2008) 'An open integrated exploratorium for database courses', ACM SIGCSE Bulletin, 40(3), pp. 22–26

[25] Brusilovsky, P. et al. (2010) Learning SQL programming with interactive tools: From integration to personalization, ACM Transactions on Computing Education (TOCE), 9(4), pp. 19

[26] Sadiq, S. et al. (2004) SQLator: an online SQL learning workbench, in ACM SIGCSE Bulletin. Leeds, United Kingdom: ACM, pp. 223–227.