Students' Syntactic Mistakes in Writing Seven Different Types of SQL Queries and its Application to Predicting Students' Success

Alireza Ahadi
University of Technology Sydney
Australia
Alireza.Ahadi@uts.edu.au

Vahid Behbood
University of Technology Sydney
Australia
Vahid.Behbood@uts.edu.au

Arto Vihavainen
University of Helsinki,
Finland
Arto.Vihavainen@cs.helsinki.fi

Julia Prior
University of Technology Sydney
Australia
Julia.Prior@uts.edu.au

Raymond Lister
University of Technology Sydney
Australia
Raymond.Lister@uts.edu.au

ABSTRACT
The computing education community has studied extensively the errors of novice programmers. In contrast, little attention has been given to student's mistake in writing SQL statements. This paper represents the first large scale quantitative analysis of the student's syntactic mistakes in writing different types of SQL queries. Over 160 thousand snapshots of SQL queries were collected from over 2000 students across eight years. We describe the most common types of syntactic errors that students make. We also describe our development of an automatic classifier with an overall accuracy of 0.78 for predicting student performance in writing SQL queries.

Categories and Subject Descriptors
H.2.3 [**Database Management**]: Languages – query languages.

General Terms
Management, Measurement, Human Factors.

Keywords
Online assessment; databases; SQL queries; Machine learning.

1. INTRODUCTION
The Structured Query Language (SQL) is the standard language for relational and object-relational databases. A better understanding of student SQL errors and misconceptions would improve the teaching and learning of SQL. It would also serve the writing of textbooks and other instructional materials. As with any other computer language, SQL queries may contain semantic or syntactic errors. The focus of the previous studies of novices in writing SQL queries has been on the nature of semantic errors. There has been little attention to analyzing the syntactic mistakes made by students when writing SQL queries.

In this paper we use data collected over eight years, from 2300 students, to quantitatively study the syntactic errors committed by students at the authors' institution. We review these mistakes among seven different types of SQL queries and compare syntactic errors of students who were successful versus students who were unsuccessful in writing a correct query. We also show how this information can be used to train a rule-based classifier to predict student success at writing an SQL query.

This paper is structured as follows. In section 2, we review the literature on analysis of semantic errors in SQL. In section 3, we describe how the data was collected and how the classifier was trained. In section 4, we review our findings on syntactic error analysis and demonstrate how the information obtained from student snapshots can be used to predict their performance in writing SQL queries under exam condition. Section 5 expands and discusses our findings. Section 6 presents our conclusions.

2. RELATED WORK
Many reports have been published about online SQL tutoring/assessment tools. However, most of those reports focus on the functionality of the tool itself, or on how the system supports a certain pedagogical model [1-6]. Those reports do not analyze the data collected by those systems to determine the syntactic errors that students face when writing SQL queries.

There are some papers in which authors review different semantic errors encountered in writing SQL queries. Reisner [8] categorized queries generated by subjects into three categories: "correct", "minor error(s) only", and "major error(s). Wetly and

SIGCSE '16, March 02-05, 2016, Memphis, TN, USA
© 2016 ACM. ISBN 978-1-4503-3685-7/16/03…$15.00
DOI: http://dx.doi.org/10.1145/2839509.2844640

Stemple [7] and used a more elaborate categorization scheme: "correct, "minor language error", "minor operand error", "minor substance error", "correctable", "major substance error", "major language error", "incomplete" and "unattempted". Wetly and Stemple studied how subjects faired at writing SQL queries in comparison to a a more procedural query language (TABLET). Later in 1985, Welty [9] ran an experiment on a small number of subjects (N=39) to test how assistance with error correction would affect SQL user performance. In that study, errors are categorized into "minor syntactic", "complex errors", "group by", "semantic error", "incorrect" and "unattempted". Buitendijk [10] categorized SQL errors into the categories of "existence", "comparison", "extension" and "complexity". In that work, a classification of natural language questions was introduced to give insight into possible errors in SQL queries. Smelcer [11] reports seven different type of common mistakes in writing SQL queries, which are "omitting the join clause", "AND/OR difficulties", "omitting quotes", "omitting the FROM clause", "omitting qualifications", "misspellings" and "synonyms". In that work, a model of query writing is developed that integrates a GOMS-type analysis of query writing with the characteristics of human cognition. Brass [12] reports an extensive list of conditions that are strong indications of semantic errors. Although all those works give a fundamental understanding of semantic errors in SQL, none of them reviewed syntactic errors.

3. METHOD

3.1 Data Collection

The data collected in this study forms a total number of ~161000 SQL SELECT statements from ~2300 students. We collected our data in an online assessment system, AsseSQL [5, 6]. The students in this study were all novice undergraduate students enrolled in an introductory database course at the authors' university. Most of the students enrolled in this course were studying for a Bachelor degree in Information Technology or Software Engineering. Each semester there is an online SQL test assessing student's performance in writing SQL SELECT statements. In the online test, students were allowed 50 minutes to attempt seven SQL questions. Each question examines the students' ability to write a SELECT statement which covers a fundamental concept. Table 1 represents these question and the total number of snapshots collected from students' attempts. Each of the seven questions presented to a specific student is chosen at random from a small pool of questions.

Table 1. Number of snapshots generated by students for different SQL SELECT statements.

Type of SELECT statement required in answer	Number of SQL statements collected from students
Group by with having	31484 (20%)
Self-join	27350 (17%)
Group by	24422 (15%)
Natural join	24248 (15%)
Simple subquery	18860 (12%)
Simple, one table	18440 (11%)
Correlated subquery	15856 (10%)

3.2 Error Categorization

In order to collect the execution result of students' attempts, all the students' SQL statements were re-executed in PostgreSQL and the execution results generated by the DBMS were collected. Based on the execution message returned by DBMS, we define a *syntactic error* as an error message which is returned by PostgreSQL engine. In contrast, a *semantic error* is produced by a successfully running query that does not produce the correct answer. For the sake of clarity, we redefine *syntax error* (Error code 42601) as a type of syntactic error which is due to either a typo or the exclusion of a semicolon at the end of the query. Table 2 reviews these error codes and their frequency among students' queries.

Table 2. PostgreSQL error codes and their frequency in student's attempts

PSQL Error code	Description	Frequency
Error 42601	Syntax error	34504 (21%)
Error 42703	Undefined column	20689 (13%)
Error 42803	Grouping error	15442 (10%)
Error 42P01	Undefined table	5548 (3%)
Error 42702	Ambiguous column	4844 (3%)
Error 42883	Undefined function	2534 (2%)
Error 42804	Data type mismatch	1262 (1%)
Error 22008	Date time field overflow	975 (0.61%)
Error 22007	Invalid text representation	849 (0.53%)
Error 22P02	Invalid date time format	536 (0.33%)
Error 3F000	Invalid schema name	457 (0.28%)
Error 42704	Undefined object	134 (0.08%)
Error 42712	Duplicate alias	108 (0.07%)
Error 42P10	Invalid column reference	96 (0.06%)
Error 42P02	Undefined parameter	47 (0.03%)
Error 42725	Ambiguous function	21 (0.01%)

In many cases, an error might arise for multiple reasons. As a result, the error code alone might not be sufficiently self-explanatory to express exactly where the problem lies within the corresponding SQL statement. Therefore, we categorized those error codes which might arise due to different reasons into sub-categories that are summarized in Table 3.

3.3 Classification

In order to investigate the predictive value of the syntactic errors of students' attempts, we trained a classifier to see to what extent this information can be used to distinguish between successful students and unsuccessful students. To that end, information on syntactic errors (i.e. the top eight syntactic errors in Table 2), the number of semantic errors and the total number of attempts of 480 students were used to build the training set. AsseSQL selects questions for each student on a random basis from a question pool. As a result, we decided to train the classifier on the proportion of students who were assigned with the same identical

question. This training set included 240 students who were successful in answering a GROUP BY question correctly (Positive set) as well as 240 students who were not able to answer a GROUP BY question (Negative set). We chose to study the GROUP BY question as it had a particularly high ratio of syntactic errors in students' attempts. The classifier trained in this study is PART, a rule based classifier which, in each iteration, builds a partial C4.5 and re-expresses the best leaf as a rule [13]. The PART classifier was selected for two main reasons. First PART handles missing values caused by student queries which no syntactic errors of a given kind. Second, PART provides a clear explanation of the rules generated within the C4.5 iterations. Feature selection and classifier evaluation was performed using the WEKA Data Mining toolkit [14].

Table 3. PostgreSQL error codes and underlying reasons.

PSQL Error code	Reason	Occurrence
42601	Wrong syntax	33482 (97%)
	Chunk of code not closed	681 (2%)
	Invalid subquery syntax	268 (<1%)
	Other rare syntactic errors	73 (<1%)
42803	Aggregate function may not be used in GROUP BY clause	13019 (84%)
	Aggregate function may not be used in WHERE clause	2294 (14%)
	Aggregate function may not be nested	104 (2%)
42883	Operator does not exist	1628 (64%)
	Function does not exist	906 (36%)
42804	Argument of AND must be type Boolean	547 (44%)
	Argument of OR must be type Boolean	437 (35%)
	Argument of HAVING must be type Boolean	271 (21%)

4. RESULTS

4.1 Syntactic Errors Have a High Frequency Ratio Among Students' Attempts

Among ~161000 students' attempts, more than half of those attempts result in a syntactic error (54%). Around 40% of all executions do not include a syntactic error, but include semantic errors. Thus only the remaining 6% of executions result in a successful execution capable of producing the correct result table. A low percentage of successful executions is to be expected, as a student stops attempting a question in the online system as soon as they have generated a correct answer. However, this low percentage of correct answers does reflect: (1) the long sequence of incorrect queries, both syntactic and semantic, typically generated by students before arriving at the correct answer, and (2) the many students who never generate a correct answer to some questions.

Table 4 reviews the frequency of incorrect SQL statements written by students. As can be seen, the number of syntactic errors in most categories is more than the number of semantic errors. Examining the last attempt of unsuccessful students revealed that half (51%) of unsuccessful students abandoned the question when they were not able to fix a syntactic error. As shown in Table 2, error 42601 ("syntax error") is the biggest category of syntactic errors. The main cause of this error is typos. The second largest category of encountered errors is error 42703 which is generated when the referenced column does not exist (which may also be caused by a typo).

Table 4. Incorrect SQL statements and syntactic errors. Second column represents the percentage of incorrect queries among all queries collected for each query type. Third column represents the percentage of incorrect queries among all incorrect queries per query type where the incorrectness is due to syntactic errors.

SELECT statement type	Percentage of unsuccessful statements	Unsuccessful due to syntactic errors
Simple, one table	89%	54%
Group by	93%	68%
Group by with having	96%	61%
Natural join	94%	66.%
Simple subquery	92%	64%
Self-join	98%	38%
Correlated subquery	93%	55%

4.2 Syntactic errors in different types of SQL SELECT statements

To better understand which syntactic errors students encounter in writing different types of SQL statements, we investigated the error codes generated by their attempts in answering seven different types of SQL questions. We chose these seven types of queries because the relative difficulty of these seven types has been studied and established by Ahadi *et. al*. [15]. As can be seen in Table 5, a limited number of error codes form the majority of the total population of most encountered errors in the seven different query types. However, the frequency of these errors varies between the seven query types. For each query type, with the exception of the self-join, more than half of the students who did not get a right answer gave up when they were not able to fix the syntactic error produced by their SELECT statement.

4.3 Successful vs. Unsuccessful: Syntactic Error Comparison

To characterize students as "successful" or "unsuccessful" according to their syntactic errors, for each SQL query type we

selected an equal number of students (N >300) and compared the total number of semantic and syntactic errors (Table 6). As an example of how this table was constructed, consider the 81% figure shown in the top left of Table 6. For error code 42601, from the total number of queries generated by successful and unsuccessful students, 81% are from unsuccessful students.

According to the information presented in Table 6, unsuccessful students tend to have more syntactic and semantic errors compared to successful students. Although the number of encountered syntactic errors differs from query type to another, the majority of syntactic and semantic errors in each error category are from unsuccessful students.

Table 5. Most encountered errors in different SQL statements.

SQL type	1st error	2nd error	3rd error	All Other Errors
Simple, one table	42601 (46%)	42703 (19%)	42803 (13%)	22%
Group by	42601 (52%)	42803 (26%)	42703 (11%)	11%
Group by with having	42601 (45%)	42803 (29%)	42703 (15%)	11%
Natural join	42703 (38%)	42601 (27%)	42P01 (12%)	23%
Simple subquery	42703 (33%)	42601 (28%)	42803 (13%)	26%
Self-join	42601 (37%)	42703 (23%)	42P01 (11%)	29%
Correlated subquery	42601 (35%)	42703 (32%)	42P01 (11%)	22%

4.4 Syntax Error Based Prediction of Unsuccessful Students

As we were able to characterize successful and unsuccessful students according to their syntactic errors, we decided to see to what extent a student's degree of struggle would be a good predictor of student's success in eventually writing the correct query. We therefore trained the PART classifier on an equal number (240) of successful and unsuccessful students in answering GROUP BY questions (i.e. N=480). On a 10-fold cross validation training mode, the classifier was able to correctly classify 77% of students correctly. Table 7 provides further details on the performance of the classifier.

To reduce the number of overlapping features, reduce over fitting, and to improve predictive accuracy of the feature set, feature selection was performed. We used correlation-based feature subset selection, where the individual predictive ability of each feature along with the degree of redundancy between the features was evaluated using three methods; (1) genetic search, (2) best first method and (3) greedy stepwise method. After feature selection, four features were used to train this classifier. Those features were: (1) how many times a student's queries contained error 42803, (2) contained error 42601, (3) a semantic error, and (4) total number of attempts student at the question.

Using those four features, five rules were generated by PART. Inspection of the rules identified features distinguishing successful and unsuccessful students. For example, two of the rules generated by our classifier are:

- If the total number of attempts is higher than 30 and there is at least one syntactic error of code 42803, then the student is not successful.
- If the student has not encountered a semantic error in answering the question, then the student is not successful. (A complete absence of semantic errors usually indicates that all the student's attempts contained syntactic errors.)

We discuss these rules, and other findings, in the next section.

5. DISCUSSION

According to our analysis, the majority of students' mistakes in writing SQL SELECT statements are either syntax errors or an incorrect referred column in different clauses of the SELECT statement. While the former seems to be a result of lack of practice, the second category gives us some insights into student's understanding. The most common mistake among students' attempts in writing a simple join query is an incorrect column name in either the SELECT or WHERE clause. In some cases, this is simply due to a typing error. In cases where the incorrect column name is not a typing error, perhaps the complexity of the provided entity relationship diagram leads students to make a mistake in choosing the right field name.

Table 6. Distribution of errors for unsuccessful students compared to successful students in different query types. The values in the table represents the proportion of unsuccessful students from the total mistake for each error and cross different query types.

PSQL Error code	Simple, one table	Group by	Group by with having	Natural join	Simple subquery	Self-join	Correlated subquery
Error 42601	81%	70%	71%	73%	70%	73%	72%
Error 42702	63%	90%	75%	70%	69%	46%	75%
Error 42703	73%	81%	68%	70%	61%	58%	72%
Error 42803	84%	69%	71%	70%	72%	73%	61%
Error 42804	89%	76%	76%	85%	77%	76%	50%
Error 42883	85%	85%	71%	74%	88%	60%	69%
Error 42P01	77%	86%	78%	65%	74%	64%	78%
semantic error	75%	61%	61%	61%	53%	50%	59%

Table 7. PART classifier's performance in classifying successful and unsuccessful students in writing a GROUP BY query.

TP Ratio	FP Ratio	Precision	Recall	F-Measure	ROC Area	Class
0.84	0.30	0.74	0.84	0.79	0.84	Unsuccessful
0.70	0.16	0.81	0.70	0.75	0.84	Successful
0.77	0.23	0.77	0.77	0.77	0.84	Weighted Avg.

We were surprised to find that error 42803 is the third most common mistake in writing a simple SELECT statement on one table, accounting for 13% of all syntax errors. When this error occurred, it almost always occurred in the WHERE clause. A simple SELECT doesn't require a GROUP BY clause, which probably indicates a serious misconception.

Inspecting the top three most encountered syntactic mistakes among seven different categories of SQL queries, error 42P01 (i.e. undefined table) is observed in four different types of queries:

a) When there is a need to perform a natural join between two relations.

b) When writing a simple subquery

c) When there is a need to join a table to itself.

d) When writing a correlated subquery.

What these four categories of queries share is the presence of either more than one table in the SELECT statement or the presence of more than one SELECT clause. This might be an indicator of students' lack of skill in identifying the correct tables from which to extract the desired information. On the other hand, investigating a small number of their queries shows that the main mistake they make is mistyping the correct name of the table, even though the number of tables used in each entity relationship diagram is small.

The results from different machine learning algorithms with the same goal can be extremely context-dependent. Even with the same algorithm, variation can even occur with variations in the exact training data used. We trained a wide range of models to classify student's performance in the online test and for each classifier, we obtained a prediction accuracies ranging from 60% to 79%. Each of these classifiers has a different set of generated rules and selected features

6. CONCLUSION

Our study goes a small way to addressing that imbalance in the computing education literature between the study of novices writing programs and novices writing SQL queries. Our data is from a relatively large data set (over 161000 SQL queries collected from ~2300 students) for a range of different syntactic errors that novices make in writing SQL statements. Our findings show that, in general, students make more syntactic errors than semantic errors. Furthermore, a syntactic error and not a semantic error is what in most cases causes a student to abandon answering a question. Syntax errors in different clauses of the SELECT statement, undefined referred column errors, and grouping errors are the most common mistakes that novices make in writing their SQL statements. The prevalence of syntactic errors, and in particular the fact that syntactic errors is what leads students to abandon attempting a question, indicates that the teaching of SQL queries needs to place greater emphasis on syntax and syntax errors. While we believe that semantic errors are the more intellectually demanding errors, students will not come to grips with semantic errors while their query formulation remains dominated by syntax errors.

7. REFERENCES

[1] Brusilovsky, P., Sosnovsky, S., Lee, D., Yudelson, M., Zadorozhny, V., and Zhou, X. 2008. An open integrated exploratorium for database courses. *ITiCSE* '08. pp. 22-26.

[2] Brusilovsky, P., Sosnovsky, S., Yudelson, M. V., Lee, D. H., Zadorozhny, V., and Zhou, X. 2010. Learning SQL programming with interactive tools: From integration to personalization. Trans. Comput. Educ. 9, 4, Article 19 (January 2010).

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

[4] Mitrovic, A. 2003. An intelligent SQL tutor on the web. Int. J. Artif. Intell. Ed. 13, 2-4 (April 2003), pp. 173-197.

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

[6] Prior, J. 2014. AsseSQL: an online, browser-based SQL skills assessment tool. ITiCSE 2014. pp. 327-327.

[7] Wetly, C. and Stemple, D. 1981. Human factors comparison of a procedural and a nonprocedural query language. ACM Transactions on Database Systems (TODS) 6;4:626-629

[8] Reisner, P. 1977. Use of psychological experimentation as an aid to development of a query language. IEEE Trans. Softw. Eng. SE-3, 3 (1977), 218-229.

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

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

[11] Smelcer, J. B. 1995. User error in database query composition. Int. J. Human-Computer Studies (1995) 42, 353-381

[12] Brass, S. and Goldberg, C. 2006. Semantic error in SQL queries: A quite complete list. The Journal of Systems and Software 79 (2006) 630–644.

[13] Frank, E. and Witten, I. H. 1998. Generating accurate rule sets without global optimization. (Working paper 98/2). Hamilton, New Zealand: University of Waikato, Department of Computer Science.

[14] Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P. and Witten. I. H. 2009. The WEKA data mining software:

an update. ACM SIGKDD explorations newsletter, 11(1):10-18, 2009.

[15] Ahadi, A., Prior, J., Behbood, V. and Lister, R. 2015. A quantitative study of the difficulty for novices of writing seven different types of SQL queries. ITiCSE (2015) Proceedings of the 2015 ACM Conference on Innovation and Technology in Computer Science Education 201-206.