

# Adaptive SQL Analysis & Refactoring System Model

Soumya Reddy  
M.Tech in Software Engineering  
Aurora's Engineering College  
Bhongir  
Nalgonda Dist-508 116, India

Srikanth Jatla  
Associate Professor and HOD of C.S.E  
Aurora's Engineering College  
Bhongir  
Nalgonda Dist-508 116, India

## ABSTRACT

In any form of education continuous learning and training plays an important role while knowledge and skills are equally important. In a given domain, an automated system for training skills and active learning is an effective form of learning. This paper presents a system that helps in active learning and training of skills with respect to database programming. While learning database programming with SQL commands, this system guides the user by giving feedback. The recommendations and the automated feedback help end user to improve his skills and focus on weak areas. Classification and correction component proposed in this paper is the core of the system which is based on syntax and grammars and provides necessary feedback. The approaches used are comparison techniques and syntax tree based abstractions which were proved effective.

## General Terms

SQL analysis, refactoring

## Index Terms

Syntax, grammar, comparison techniques, artificial intelligence, education, query language

## 1. INTRODUCTION

In any active learning system, feedback and suggestions are important processes that can guide end user to learn skills and gain knowledge faster. It is true of such system in any domain. However, the feedback should be precise, relevant and understandable. In conventional learning a student is associated with his teacher and gets accurate feedback as his teacher acts as guide with sufficient domain or subject knowledge in the process of learning actively. However, in a computer-aided application, it is imperative that the system should act as teacher which needs some sort of artificial intelligence. From this kind of system student gets help as it is available round the clock. This is very useful as human tutors may not be available always [7]. For such interactive learning systems, formally defined languages are useful as they have specific syntax and semantics. One such language for database interaction is SQL.

SQL is widely used query language to interact with relational databases. This language is simple English like language with simple syntax. Its complexity is medium in terms of its grammar [8]. The main objective of this paper is to investigate an integrated approach to have features like domain specific feedback, correction and personalized feedback. A correction technique is the core of the whole work.

- This paper develops techniques to analyze given SQL select query and identify syntax or other problems generally encountered by students while learning and working with database through SQL.
- Knowledge based interactive and feedback system which is adaptive is the main focus of this paper.

This paper introduces the required data structures and techniques for analyzing SQL select query in order to give personalized feedback to end user which helps him learn from given feedback. The error correction technique which is pattern-based is introduced here. It is responsible to identify student weakness and suggest further learning as well. Since it is syntax driven approach a syntax tree abstraction is also used as part of this solution. Apart from this two important problem identification and correction techniques are introduced.

- Equivalence rules
- Pattern matching

The equivalence rules are used to find semantically equivalence while pattern matching technique is used to localize and classify errors.

## 2. RELATED WORK

The proposed learning and training system is based on the architecture of ITS. Three systems are introduced from literature that support learning and training, especially personalized feedback for SQL tutoring. These systems are similar in interface and aims to some of the existing ones. However, they differ in terms of methods of error correction and architecture:

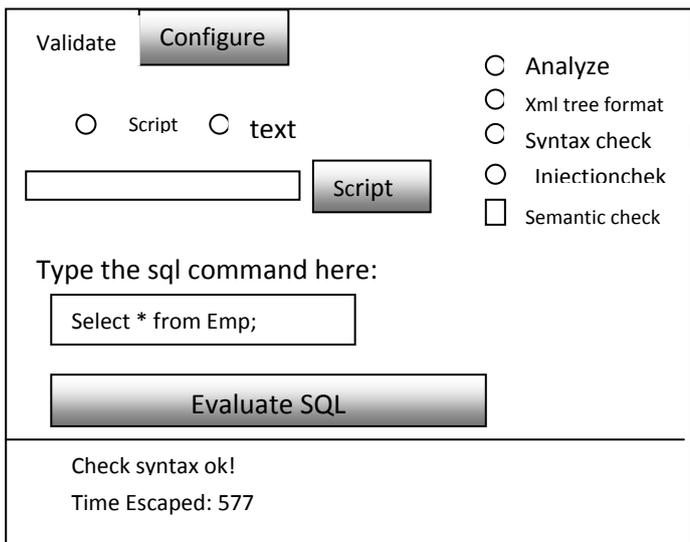
- SQLator [15] is a student submission evaluation framework which equates the submission with corresponding English question. It examines an English question and corresponding SQL SELECT query, analyzes it and judges if the answer in the form of SQL query is correct without actually executing the SQL query.
- Acharya [3] introduces a three step process in order to correct given answer of student. The three steps are namely pre-processing, atom processing and truth table processing. In terms of syntax-driven approach, it is similar to the approach of this paper. It has some assumption that the literals of two language expressions are same. This system fails if one of the language expressions has more literals of atoms. Thus it has accuracy problems. In its present work, only where clause of given SELECT query is analyzed. The truth table technique is used to determine the equivalence between two language expressions.

- SQL-Tutor [11] used different approach in solving the same problem. It uses constraints based reasoning for correcting queries submitted by student. Here the constraints are stored in a constraints base which is used to deal with syntax and semantic errors. This system verifies given query with constraints for syntax and semantics without actually executing SQL SELECT query. This system has produced high level of accuracy. However, its success depends on the quality constraint base and the level of its sophistication. In order to support large number of database errors and correction, this approach has to continuously and incrementally update its constraints base periodically.

In terms of giving feedback to student the first system makes use of an error flag which signals the correctness of the answer. Error flagging and hints are used in the second system for the same while the third system described above provides advanced feedback comprising partial solutions and hints. A suggestion for next question is suggested by both Acharya and SQL-Tutor. However, the reason for this kind of approach is not explicitly provided.

### 3. THE PROPOSED FRAMEWORK

The proposed system is known as automated SQL Tutor. It is part of the IDLE (Interactive Database Learning Environment). Its main user interface is as shown in fig. 1.



### 3.1 Learning and Training

Interaction which is knowledge-level is the success factor in the interactions between student and the interactive learning systems [14]. Any Intelligent Tutoring System (ITS) has to support features of SQL programming as part of its online exercises [9]:

- It provides a wide variety of SQL problems.
- It can correct student answer while giving right feedback.
- At the end of each lesson, it can give personalized recommendations.

### 3.2 Pattern Matching and Intelligent Tutoring

ITS plays an important role in interactive learning. ITS has a variety of strategies to support students in interactive learning [12]. They are very effective and earlier systems were restrictive that could limit student's learning experience [5]. Any conventional ITS has four distinct parts [4]. They are the expert model, the student model, the pedagogical model and the interface. The expert model has knowledge. The student model has student information along with his knowledge levels. The pedagogical model determines when and how to instruct student properly while the interface has means of communication between ITS and student.

Pattern matching is a technique used in interactive learning systems in order to define ideal learning paths and solutions [12]. CAPIT, TAO are two existing systems that make use of pattern matching method [10][16].

### 4. ARCHITECTURE OF THE PROPOSED SYSTEM

A two layered model is used to represent the information architecture of the proposed system. They are content model and student model. The former captures language expressions in the form of syntax trees for further analysis while the latter is for capturing feedback preferences given by student and his performance. The architecture of the proposed system is as shown in fig. 2.

As can be seen in the above architecture, the system has four components namely an interface, a student component, a correction component and a guidance and recommendation component. The interface component is responsible to provide necessary interaction to the student for executing SQL commands and learning from the feedback given. The student component represents student model, correction component is responsible to verify student query for given problem and give suggestions or feedback while the guidance and recommendation component is meant for assessing student's week areas and giving recommendations to improve it.

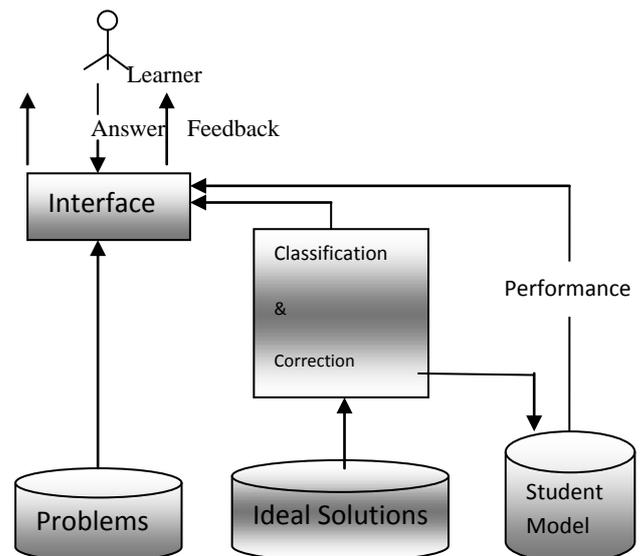


Fig. 2 shows architecture used for error correction

## **5. LANGUAGE REPRESENTATION**

The computer languages here are considered as the area of learning as well as training have been computer-process able formal definition given languages for the design, modeling and implementing computer systems.

The three available facets relative to computer languages are discussed below [1]:

Lexical as well as structural aspects are referred by syntax. On the basis of a vocabulary involving keywords and elements that are user defined, formulation of sentences in a particular language can be done. These sentences can be constructed under the governance of a grammar.

Meaning relative to elements as well as sentences of a language can be referred by Semantics. This happens usually in a mathematical manner or through mapping onto remaining notations.

The language use is referred by Pragmatics. It is used to capture conventional rules as well as guidelines that, for example, enhance the readable nature of a sentence or the expression quality.

Concrete syntax is related with the original representation as it is offered by the user which means through keywords and various lexical concerns like the identifier construction.

Validation of Concrete syntax is generally encouraged by processors of language

like syntax checkers or execution tools or compilers [1]. This aspect is ignored. Here the abstract syntax is concerned, that may be given definition in grammars terms. Abstract syntax trees or ASTs are used to represent sentences that are grammatically correct.

Abstract Syntax Trees being representations relative to language expressions change the center of attention from lexical concerns towards structural notational ones [1],[2]. In certain cases, additional abstractions of an Abstract Syntax Tree can happen for facilitating the application of particular transformations or analysis. For example, tree elements can be grouped and classified.

## **6. INTERACTIVE LEARNING**

The technique of correction is able to provide immediate synchronous feedback to learners so that they can submit to the tutoring system of SQL. Learning as well as training is, moreover, incessant processes above a list of individual interactions.

The student is supported individually via observation as well as personalized feedback and recommendation has been one major advantage of computer-enhanced learning [6]. Feedback could be either local or global [9] that are addressed by synchronous local feedback that has been part of corrections, global guidance and recommendations depending on the overall performance of the student. The correction is revisited but from the perspective of interaction while considering the student model as well as feedback generation, the feature of recommendation is introduced.

## **7. ANALYZING RESULTS**

### **7.1 Accuracy**

Accuracy is defined as the important property of a correction system: if it can be used only for feedback or also for grading. The extent of accuracy may be made vulnerable via two factors in case of an approach that is syntax-driven.

Determination of correctness: The possibility to have solutions that are semantically equivalent and different in a syntactical manner can have its impact on accuracy in a negative manner. An overall set of rules of equivalence should be defined and even applied.

Classification of errors: Accuracy can be affected by the complex nature of the language itself and correctness can still be easier for dealing with, the quality of various categories of error may be complex for addressing in a system used for weighting as well as ranking for multiple number of errors.

It has been a very complex issue in achieving accuracy practically. The SQL query language is having medium complex nature which is even more complex when compared to certain diagrammatic notations while it is of simple nature when compared to certain programming languages which integrate various computational paradigms as well as mechanisms.

### **7.2 Evaluation**

The SQL tutor which is described has been implemented. Student attainment as well as student opinion are two main achievement criteria to learn support tools. An assessment of behavior of student in the systems can also include important information in terms of system efficiency.

A method of hybrid evaluation depends on conventional surveys and a novel kind of data mining. Conventional information about behavior of student and system usage can be decided via Web usage mining, which here is used for validation and for complementing the results of the evaluation of attainment and survey of students. Even though this evaluation represents examples regarding just-in-time learning, it is seen that this kind of behavior of learning is complemented through long-term as well as preemptive usage. The major part of usage happened beyond the laboratory sessions that are scheduled, which gives demonstration of its value as a tool of self-study following the philosophy of apprenticeship.

Student attainment is considered as a crucial metric for determining the achievement of e-learning system. Moreover, the motivation or organization or system usage by the students constitutes main factors that decide the acceptance and final attainment.

### **7.3 Adapting to Other Languages**

The goal is to provide a standard approach which may be applicable to various computer languages. This could highly involve languages that are diagrammatic and also programming languages that more complex.

Applying the approach towards complex programming or specification, or modeling language needs little effort of implementation for achieving acceptance at least level, languages having same or smaller complex nature could be addressed through practical terms.

A language represented in grammar terms, ASTs as well as SASTs, and equivalence rules of semantics depending on the grammar may be only preliminary requirement. Grammars formulated in a suitable manner and permit the efficient

parsing of an expression of language are indeed required. After the language gets mapped against this infrastructure, the proposed method can be used, but limited only by the scalable nature of the problem of accuracy.

## 8. CONCLUSION

The proposed automated learning system is capable of providing a realistic learning environment for database programming. It is location and time independent which has two knowledge data representation techniques.

\* Language grammars for defining the learning content and knowledge.

\* Syntax trees for representing problem based knowledge.

The system demonstrated the advanced tutoring for a computer language, the effectiveness of pattern-based correction for personalized feedback and guidance.

The students who have subjects related to database programming can get benefited from this system. They can submit queries and get feedback and guidance by the system. Developing a flow less correction method is a challenge. However, the experiments demonstrated that for languages like SQL sufficient accuracy can be achieved. Nevertheless, an important note here is that an automated learning system can never replace a human tutor in terms of quality.

## 9. REFERENCES

- [1] A.V. Aho, M.S. Lam, R. Sethi, and J.D. Ullman, *Compilers: Principles, Techniques, and Tools*, second ed. Addison-Wesley, 2007.
- [2] A.V. Aho, J.D. Ullman, and J.E. Hopcroft, *Data Structures and Algorithms*, series in Computer Science and Information Processing. Addison-Wesley, 1983.
- [3] S. Bhagat, L. Bhagat, J. Kavalan, and M. Sasikumar. Acharya, "An Intelligent Tutoring Environment for Learning SQL," Proc. Int'l Conf. Online Learning (Vidyakash), 2002.
- [4] T. Boyle, *Design for Multimedia Learning*. Prentice Hall, 1997.
- [5] P. Brusilovsky, "Adaptive Hypermedia: From Intelligent Tutoring Systems to Web-Based Education," Proc. Fifth Int'l Conf. Intelligent Tutoring Systems (ITS '00), pp. 1-7, 2000.
- [6] P. De Bra, D. Smits, and N. Stash, "Creating and Delivering Adaptive Courses with AHA!," Proc. First European Conf. Technology Enhanced Learning (EC-TEL '06), 2006.
- [7] D. Heaney and C. Daly, "Mass Production of Individual Feedback," Proc. Ninth Ann. ACM SIGCSE Int'l Conf. Innovation and Technology in Computer Science Education (ITiCSE '04), pp. 117-121, 2004.
- [8] C. Kenny, "Automated Tutoring for a Database Skills Training Environment," MSc thesis, School of Computing, Dublin City Univ., 2006.
- [9] C. Kenny and C. Pahl, "Automated Tutoring for a Database Skills Training Environment," Proc. ACM SIGCSE '05, pp. 58-62, 2005.
- [10] M. Mathews and A. Mitrovic, "Investigating the Effectiveness of Problem Templates on Learning in Intelligent Tutoring Systems," Proc. 13th Int'l Conf. Artificial Intelligence in Education (AIED '07), pp. 611-613, 2007.
- [11] A. Mitrovic, B. Martin, and P. Suraweera, "Intelligent Tutors for All: The Constraint-Based Approach," IEEE Intelligent Systems, vol. 22, no. 4, pp. 38-45, July/Aug. 2007.
- [12] T. Murray, "Authoring Intelligent Tutoring Systems: An Analysis of the State of the Art," Int'l J. Artificial Intelligence in Education, vol. 10, pp. 98-129, 1999.
- [13] C. Pahl, R. Barrett, and C. Kenny, "Supporting Active Database Learning and Training through Interactive Multimedia," Proc. Ninth Ann. Conf. Innovation and Technology in Computer Science Education (ITiCSE), 2004.
- [14] R. Ramakrishnan and J. Gehrke, *Database Management Systems*. McGraw-Hill, 2003.
- [15] S. Sadiq, M. Orłowska, W. Sadiq, and J. Lin, "SQLator An Online SQL Learning Workbench," Proc. Ninth Ann. ACM SIGCSE Int'l Conf. Innovation and Technology in Computer Science Education (ITiCSE '04), pp. 223-227, June 2004.
- [16] R.H. Stottler and M. Vinkavich, "Tactical Action Officer Intelligent Tutoring System (TAO ITS)," Proc. Interservice/ Industry Training, Simulation and Education Conf. (I/ITSEC), 2000.

## 10. AUTHORS PROFILE

Mrs. Soumya Reddy received her B.Tech in information technology from JNTU Hyderabad. Currently, she is an M.Tech student specializing in software engineering, in the department of computer science engineering, from Aurora's Engineering College, Bhongir, Andhra Pradesh, India. Her areas of interests are data mining and knowledge discovery, especially automated learning systems.

Mr. Srikanth Jatla, working as an associate professor, head of the department of computer Science & engineering at Aurora's Engineering College with a teaching experience of 12 years. He is a B.E and M.Tech in computer science and is pursuing his PhD in Data Stream Mining at JNTU, Hyderabad. His area of interest includes data structures, principles of programming languages, algorithm analysis and compiler design.