

The Generation of E-Learning Exercise Problems from Subject Ontologies

Edmond Holohan*, Mark Melia, Declan McMullen, Claus Pahl
Dublin City University, School of Computing,
Dublin 9, Ireland.
eholohan@computing.dcu.ie

Abstract

The teaching/ learning of cognitive skills, such as problem-solving, is an important goal in most forms of education. In well-structured subject areas certain exercise problem types may be precisely described by means of machine-processable knowledge structures or ontologies. These ontologies can readily be used to generate individual problem examples for the student, where each problem consists of a question and its solution. An example is given from the subject domain of computer databases.

Keywords: e-learning, ontology, databases, sql, problem generation.

1. Introduction

The content of educational courses commonly exists in the form of text and other types of document. However, most teachers will be aware of a more formal view of educational content, which models this content in terms of concepts and skills, i.e., as *declarative* and *procedural knowledge*. The learning goals of any particular course may be explicitly and clearly stated in terms these latter representations. Clear statement of learning goals facilitates subsequent evaluation of student attainment and of the course itself.

Declarative knowledge is the conceptual content of a given subject, course or lesson. It includes at least the most important concepts of the subject matter together with the various types of relationship between these concepts. It should also include examples of these concepts and relationships.

By *procedural knowledge* we mean the mainly cognitive skills required to be developed and applied by the student during a particular course. Generally it is not sufficient that a student knows about any such skill or procedure; the student must demonstrate the ability to use the skill in practice.

We and others have shown that it is possible to semi-automate the generation of static information-based e-learning content from declarative knowledge representations or ontologies [1]. Since this is accomplished by means of domain-independent algorithms only, it is applicable to any subject for which declarative knowledge representations are available. This is the current state-of-the-art.

In this paper we examine the possibility of using knowledge representations to automate the dynamic production of e-learning content for procedural knowledge/ skills. Specifically we consider the generation of e-learning exercise problems from domain-specific subject ontologies.

The remainder of this paper is organised as follows: first we explain the basic idea by means of a simple example; next we report on the automated generation of e-learning exercise problems in the domain of computer databases; we review some applications which are complementary to ours; finally, we draw some conclusions and indicate future directions.

2. The idea

The teaching/ learning of problem-solving skills requires plenty of practice, lots of varied examples, individual attention and immediate feedback. It would be helpful if some aspects of this could be automated.

Some initial headway may be made in this direction by limiting our focus to well-structured problems and simple skills, e.g., arithmetic operations. Many readers will be familiar with simple mathematics tutoring programs which exercise the student's ability with addition, multiplication, etc. The fundamental characteristics of these systems are as follows:

- the automated generation of any number of random exercise problem instances;
- the automated generation or calculation of the/a corresponding correct response(s).

What if a similar strategy could be applied to other domains?

The domain of computing languages is a well-structured one. Problems in this area usually require students to write programs or program fragments, which are correct both syntactically (i.e., they obey the rules of the language) and semantically (i.e., their output satisfies the meaning of the problem).

In the computing subject of (relational) databases there is a language known as *structured query language (SQL)*. This allows the user to specify the information s/he wishes to retrieve from database storage. A relational database organises its information in tabular form, e.g., table, *persons* (Table 1).

Table 1. persons

<i>person_id</i>	<i>person_name</i>
1001	Tom
1003	Dick
1007	Harry

For example, the problem of requesting all the values in column, *c*, of table, *t*, may be expressed by the SQL formula, *SELECT c FROM t*. Given a particular database structure (schema) with certain tables, a computer program, picking from these tables and their columns at random to provide values for *t* and *c*, may generate many instances of this problem type, e.g., *SELECT person_name FROM persons*. For tutoring purposes the program could present the problem in English, *List the names of all the persons.*, and it could request the student to enter the SQL-query expression. If a reasonably small database is used, the student could be asked to predict the resulting query output, *Tom Dick Harry*.

In our test implementation we have applied this idea to queries of moderate complexity and the user can supply his/her own database schema as input.

The efficient authoring of e-learning content is a central problem of courseware engineering. Our proof-of-concept system, OntAWare, employs the knowledge-oriented technology of the Semantic Web to generate adaptive e-learning content. The basic functionality of OntAWare is the automatic generation of a variety of simple learning objects such as slideshows and objective tests. Here we concentrate on the system's more advanced functionality, i.e., dynamic problem generation.

3. The automatic generation of exercise problems for e-learning

We describe the generation of e-learning problem-solving exercises. We focus on the subject domain of relational databases (RDBs) and specifically begin with the simple problem category of writing structured-query-language (SQL) queries.

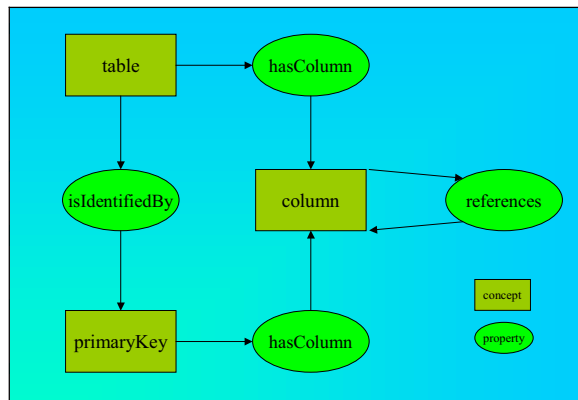


Figure 1. Simple ontology for RDB schemas

Relational ontologies are being used as a meta-language for federating multiple databases. Our system exploits these ontologies for e-learning problem generation. OntAWare generates random SQL queries from any given RDB schema. These queries are converted to an English form and posed as problems for the student.

Our work is based on that of de Laborda et al [2], who report that they are able to uniformly describe and share the schema of virtually any relational database. They obtain a model with three layers: the *Relational.OWL* layer, the schema representation layer, and the data representation layer.

Figure 1 is a diagram of the topmost level, *Relational.OWL*, slightly modified. This is the basic ontology used to represent RDB schemas. (*OWL* [3], the *Web Ontology Language*, is the preferred ontology language of the Semantic Web.) The user must first create or import the above relational schema ontology in OntAWare. We use this ontology to hold the database structure supplied by the user.

The second layer stands for ontology instance data, which are created using *Relational.OWL* in order to represent the schema of a specific relational database. The user must populate this ontology with particular schema instance information, i.e., table and column names, primaryKeys, etc. This structure forms the setting against which query examples must be generated. We do not need to use the third layer, since the queries are generated solely from the second (schema) layer.

The SQL queries, i.e., problems and solutions, are generated randomly on demand from the above

relational schema ontology and instance information. The algorithm employed may be described as a *random schema walk*. The random generation process applies both to the specific query type as well as to the query details.

```

Schema:
teachers ( teachercourse, teacherlevel, teachername, teacherteamname* )
teams ( teamdivision, teamname, teamskill )

RDB Problem:
Show the teacherlevel, teacherteamname, teachercourse, teachername and
teamskill of each of the teachers whose teachername is 'edmond' and whose
teachercourse is 'databases'.

SQL Query:
SELECT teacherlevel, teacherteamname, teachercourse, teachername,
teamskill FROM teachers, teams WHERE teacherteamname=teamname AND
teachername='edmond' AND teachercourse='databases';

```

Figure 2. Generated SQL exercise problem

Figure 2 is a sample screenshot of the system output showing a randomly generated exercise problem and solution for a given database subschema. This problem requires the student to formulate the SQL for a *join* query, since the data is to be drawn from two tables.

4. Complementary applications

IDLE [4], is a Web-based intelligent tuition system (ITS) that is currently in use for a second-year undergraduate course in a computing degree at DCU. IDLE has an adaptive, interactive lab application for learning database querying. Currently this is loaded with static pre-defined query exercises. These may now be supplemented with the query exercises produced dynamically by the OntAWARE service. The OntAWARE system is being extended to allow for pre-specification of the required query category.

SQL-Tutor [5], is based on Constraint-Based Modeling, an approach to student modeling which concentrates on violations of the basic principles in the domain of instruction.

Like IDLE, but unlike OntAWARE, it appears that SQL-Tutor must be loaded with static pre-defined databases and query exercises. However, SQL-Tutor has an impressive degree of adaptivity to the student. In the future we expect OntAWARE to dynamically produce query exercises which are adaptive to the current user's needs.

5. Conclusions and Evaluation

Our current experimentation described in this paper shows that, for at least some types of problem-solving skills acquisition, e-learning objects may be generated by domain-dependent algorithms.

The automatic generation of e-learning exercise problems, has a dual role: for the learner, it can be configured as an independent online service to produce problems on demand; for the instructor or content author, it can be used as an assistant to expedite the composition of exercise problems.

The quality of the SQL-query problems currently generated by the system is not uniform: while they are invariably syntactically correct, they are not all of equal semantic value, i.e., they are not all realistic in the ordinary sense. Typically an instructor would like to be selective about those queries employed.

Some extensions being considered for the system are the following: comparison of student's attempted solution (query) with 'correct' one; a marking scheme must be generated: this may consist of positive marks for essential phrases of the query and negative marks for irrelevant phrases; exploration of the applicability of a similar approach in different domains.

6. References

- [1] E. Holohan, M. Melia, D. McMullen and C. Pahl, Adaptive E-Learning Content Generation based on Semantic Web Technology, *Workshop on Applications of Semantic Web Technologies for e-Learning (SW-EL@ AIED'05)*, July 18, 2005, Amsterdam, The Netherlands.
- [2] C.P. de Laborda, and S. Conrad, "Relational.OWL - A Data and Schema Representation Format Based on OWL", *Proc. Second Asia-Pacific Conference on Conceptual Modelling (APCCM2005)*, Newcastle, Australia. CRPIT, 43. Hartmann, S. and Stumptner, M., Eds., ACS, 2005, pp. 89-96.
- [3] W3C, *OWL Web Ontology Language Overview W3C Recommendation*. 10 February 2004. D.L. McGuinness and F. van Harmelen (Eds.). <http://www.w3.org/TR/owl-features> (last visited on 24-Jan-2006).
- [4] C. Pahl, R. Barrett and C. Kenny, (2004). Supporting Active Database Learning and Training through Interactive Multimedia. *International Conference on Innovation and Technology in Computer Science Education ITiCSE'04*. Leeds, United Kingdom. ACM
- [5] A. Mitrovic, S. Ohlsson, Evaluation of a Constraint-based Tutor for a Database Language, *Int. J. on Artificial Intelligence in Education*, 10(3-4), (1999) 238-256.