Automatically assessing free-form diagrams in e-assessment systems

Conference or Workshop Item

Automatically assessing free-form diagrams in E-assessment systems

Dr Pete Thomas, Dr Kevin Waugh & Dr Neil Smith
Computing Department
Open University
Milton Keynes, MK7 6AA
p.g.thomas@open.ac.uk k.g.waugh@open.ac.uk n.smith@open.ac.uk

Abstract

In this paper, we describe the advances we have made in extending the quiz component of our VLE (Moodle) to support the drawing and automatic assessment of free-form diagrams. The paper illustrates how diagram-based questions are incorporated into normal Moodle quizzes and the nature of the feedback provided. It describes the types of diagrams that can be supported by the system (graph-based) and outlines how the automatic marking is performed. We briefly summarise the results of our tests of the marking algorithm which show that it performs well. Finally, the paper outlines two important support applications: one for specifying the type of a diagram to be used in a quiz and one for creating questions, model solutions and marking schemes for diagram questions.

Keywords

Free-form diagrams, automatic marking, e-assessment

1. Introduction

Diagrams play a crucial role in almost all disciplines for recording information, explaining concepts and designing solutions to problems. Unfortunately, for a variety of reasons, their use in e-assessment has been limited. Where they are available they usually occur within closed answer questions. That is, they appear in questions where there is a limited number of possible answers; effectively multiple choice questions (MCQs). They are often restricted to cases where the user can drag-and-drop onto hotspots or complete the diagram from a list of prepared elements: similar to textual fill-in the blank style questions. We have, therefore, been investigating the extent to which free-form diagrams might be incorporated into our e-assessment system [Smith et al., 2010].

A free-form diagram is a diagram drawn with few constraints on what might be constructed (although there may be limitations on the kinds of diagrammatic elements that might be used). That is, in a given discipline, a particular type of diagram is constructed from a specified set of basic elements that have specific meanings within that discipline, but there is little or no restriction on how those elements might be combined to produce a diagram. In such an environment, there is unlikely to be a small fixed number of diagrams that could be the basis for MCQs. We want to provide students with the freedom to express themselves.

In section 2 of this paper, we illustrate how we have extended the capabilities of the quiz component of our virtual learning environment to include free-form diagrams in student responses. Section 3 describes the range of diagrams that our system is capable of supporting, and section 4 summarises the experiments we have performed to assess the correctness of our automatic marking engine.

2. Extending the VLE

The Open University (OU) has, for some time, used Moodle as it's virtual learning environment (VLE) with every module having an online presence. Many modules utilize Moodle's e-assessment system both for summative and formative purposes. The OU also supports an in-house e-assessment system known as OpenMark which provides a wider range of question types than Moodle. However, we have successfully interfaced OpenMark with Moodle so that OpenMark questions can be incorporated into a Moodle test with no discernable difference to the student user. Effort is now being put into developing Moodle's quiz engine so that it can support the extensive range of OpenMark question types without having to delegate work to the OpenMark system. At the time of writing, diagramming questions are graded by OpenMark but rendered in Moodle. In this section we illustrate Moodle's user interface for diagramming questions as used on OU module M359. Figure 1 shows how the student would see the question in the Moodle space.

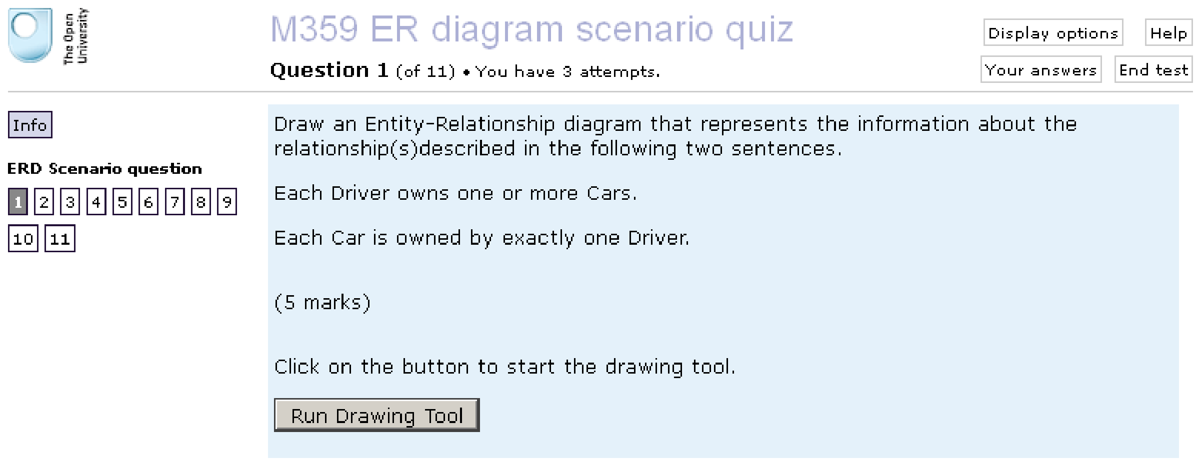

Figure 1 A diagramming question

The question has a button labelled 'Run Drawing Tool' which, when pressed, launches an applet for drawing entity-relationship (ER) diagrams required by the question and illustrated in Figure 2.

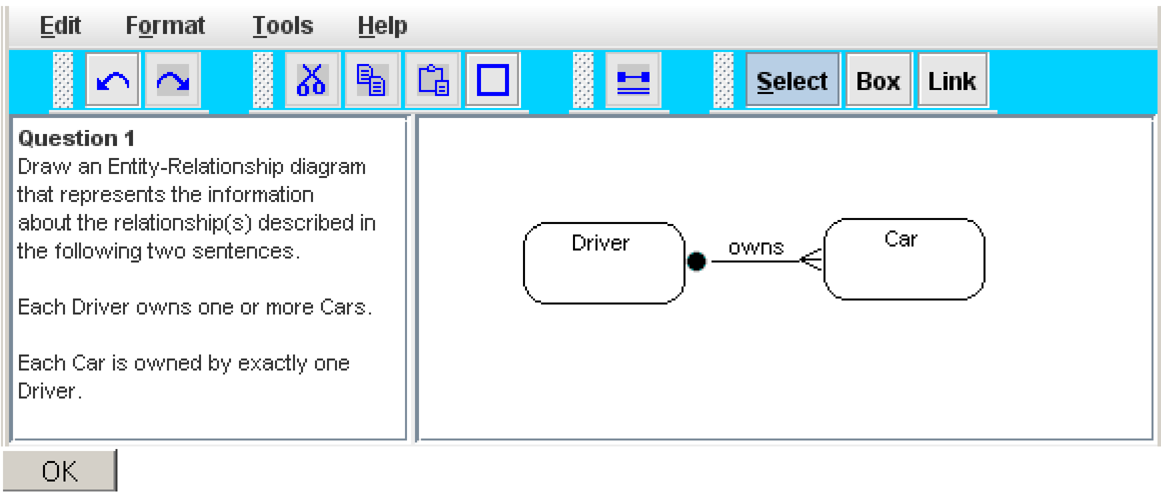

Figure 2 The drawing tool configured for ER diagrams.

The applet displays the question and provides a canvas on which to draw a diagram. Once satisfied with their diagram the student presses the OK button. The diagram is then sent to the marking engine which analyses the diagram and, in formative mode, returns a mark and feedback to the student as illustrated in Figure 3.

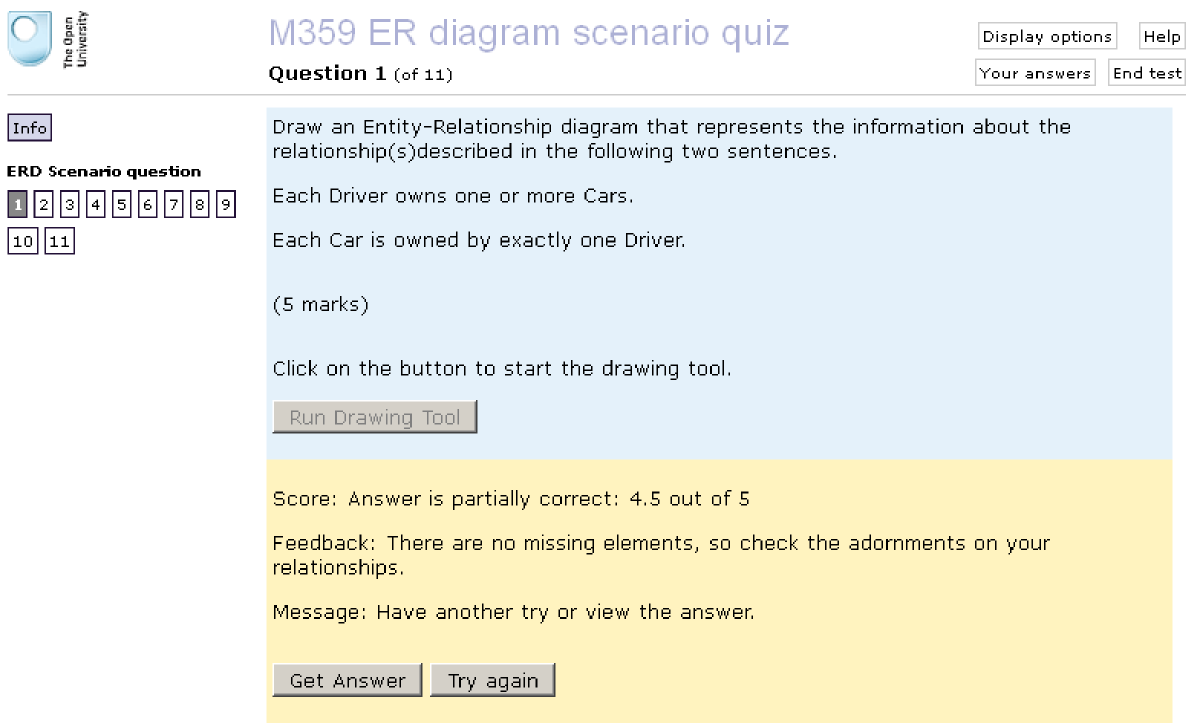

Figure 3 The response from the marking engine

At this stage the student can choose to have another attempt if their answer was not entirely correct, examine the model answer or move to another question. If the student chooses to examine the model answer, they will be presented with an interactive version of a correct diagram which can be interrogated to provide additional feedback.

In summative mode, the grade and feedback for a question are not returned to the user but are stored in a student record for subsequent assignment grading purposes.

3. Types of Diagram

Our current system is designed for structured diagrams which, in general terms, show relationships between objects of interest. Objects are usually represented by two-dimensional figures such as rectangles and circles (boxes), and relationships are represented by lines joining related objects (links). There is a surprisingly wide range of diagrams that conform to this structure with a selection shown in Figures 4 and 5.

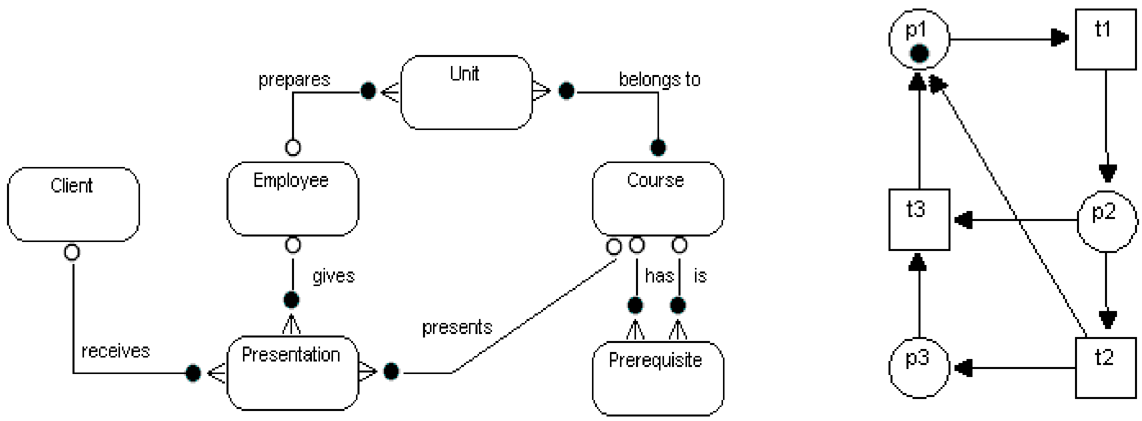

Figure 4 An ER diagram and a Petri net.

All these examples illustrate that boxes and links can have labels. The ER diagram in Figure 4 illustrates that links can have adornments in the form of icons at different positions on the lines which represent different aspects of the semantics of the relationships. The Petri net represents transitions (t_i) from one place (p_i) to another; the transitions taking place in the direction of the arrows.

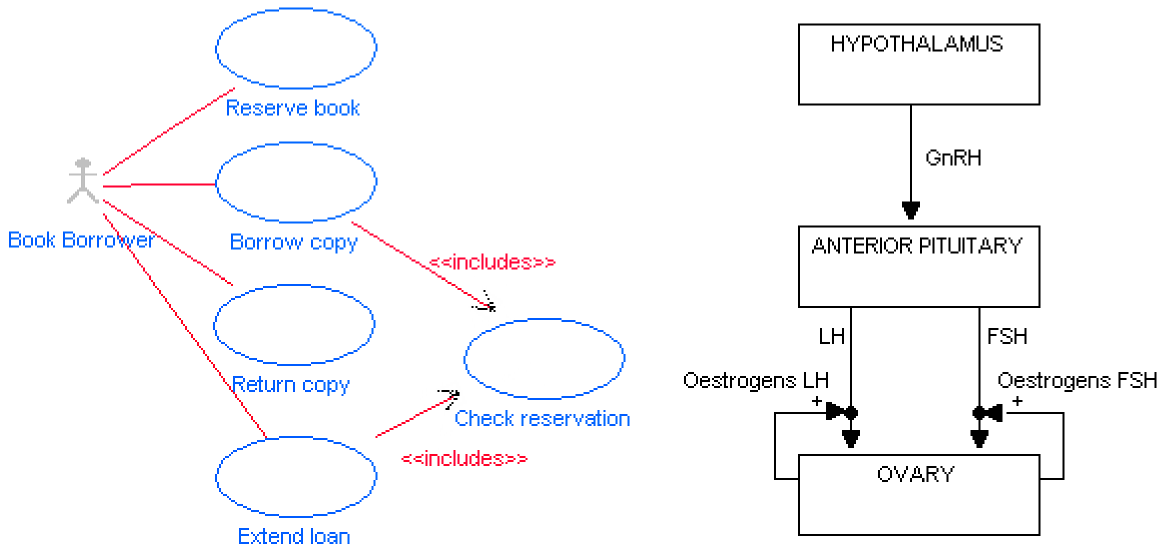

Figure 5 A Use case diagram and a Biological flow diagram

The use case diagram in Figure 5 illustrates that objects can be represented by icons, here a ‘stick’ figure, rather than boxes and that colour may indicate some semantics of the relationships. The Biological flow diagram in Figure 4 represents the flow of hormones between various glands with the flows taking place in the direction of the arrows. This diagram also illustrates that flows (or relationships) can merge together.

Generally, therefore, we have a system that supports diagrams in the form of graphs where the nodes can be boxes or icons and the edges can have adornments. Whilst not shown here, nodes can also have attributes.

4. Automatic marking

Perhaps the most important development is the ability of the marking engine to be able to accurately grade arbitrary diagrams. The method we employ has been discussed in detail elsewhere, see for example [Thomas et al. 2008] but here is a brief overview of the technique and a summary of the results of experiments to determine the correctness of the approach.

We first expect there to be at least one, or possibly more, model solution for each question (this allows for multiple acceptable solutions). Each model solution diagram is accompanied by a marking scheme that specifies how marks are to be awarded for that solution. A matching algorithm is used to compare a student's answer diagram with the model solution diagram(s). The algorithm attempts to match both nodes and edges of the abstract student answer graph with the nodes and edges of the abstract model solution graph, taking into account the attributes of nodes and adornments on edges. The aim is to match every node and edge in the answer diagram with a node and edge in a model solution. The matching process computes a similarity measure (a value in the range 0 to 1) for each pair of nodes and each pair of edges (one in the answer and one in the solution) and chooses the pairs which maximise the sum of the similarities across the two diagrams. This produces a similarity measure for the two diagrams. In cases where the similarity between two nodes or two edges is very high, the algorithm will consider the match to be 'exact' and in cases where the similarity is low the algorithm will consider the pair to be entirely unrelated. This approach is quite 'forgiving' and is akin to giving the 'benefit of the doubt', a well-known concept in grading!

Once two diagrams have been matched and similarity measures determined, a marking scheme can be applied. For example, if a number of marks are to be awarded for each correct edge, the similarity measure is used to award partial marks for an answer edge that only partially matches a solution edge. This approach has proven to be quite robust and parallels what human markers often do.

We have applied this approach to two large corpora of diagrams produced by real students in invigilated examinations. A team of three markers marked every student answer and ultimately agreed on the mark that ought to be awarded in each case; the gold standard. The marks generated by the automatic marker were compared with the gold standard and the results are presented in Table 1.

Corpus	N	Difference	0	0.5	1	1.5	2
1	394		389	4	1	0	0
		Cumulative %	98.7	99.7	100	100	100
2	117		108	6	3	0	0
		Cumulative %	92.3	97.4	100	100	100

Table 1 Experimental results

Table 1 shows that, for the first corpus of 394 diagrams, the automatic marker agreed with the gold standard in 98.7% of cases; for the second corpus, the exact agreement was lower at 92.3%. In both corpora the diagrams were marked out of 7 so we tend to quote the percentage of diagrams that were marked with no more than a half mark (7%) difference as our measure of accuracy giving 99.7% and 97.4% for the two corpora respectively. It is interesting to note that the worst performance in both corpora was only one mark difference.

5. Support systems

There are two additional applications that support the creation and use of diagrams in our VLE. The first enables the teacher to create a specification for the type of diagram to be tested. A diagram specification specialises both the drawing tool and the automatic marker for a specific type of diagram (ER diagrams and Petri Nets are two distinct types of diagram). This application can be used to define new types of graph-based diagrams.

The second application is an authoring tool that enables the teacher to construct a question (text), one or more model solutions (diagrams) and accompanying marking scheme using one of the defined diagram types. The outputs from this application are two files: one is a question file that enables the VLE to render the question in a quiz, the other contains information used by the marking engine. Further details of these applications can be found in [Thomas et al,. 2007].

6. Conclusions

This paper has demonstrated how the Open University has incorporated free-form diagram questions into its virtual learning environment. Modules are currently using the facility. We have performed a number of experiments to verify that the automatic marking engine is sufficiently accurate at least for formative purposes.

7. References

[1] Smith, Neil, Pete Thomas, and Kevin Waugh, (2010) Diagram Interpretation and e-Learning Systems, in A.K. Goel, M. Jamnik and N.H. Narayanan (Eds): Diagrams 2010, LNAI 6170, pp. 331-333. Springer, Heidelberg (2010)

[2] Pete Thomas, Neil Smith, and Kevin Waugh (2008) Automatically assessing graph-based diagrams. J. Learning, Media and Technology, 33(03), pp. 249 - 267 [Special Issue: Reframing E-assessment: Adopting New Media and Adapting Old Frameworks]

[3] Thomas, P.G., Waugh, K., and Smith, N. (2007) Tools for learning and automatically assessing graph-based diagrams. In Research Proceedings of ALT-C 2007, Nottingham, 4-6 September, 2007, pp.61-74.