



This item was submitted to Loughborough's Institutional Repository (<https://dspace.lboro.ac.uk/>) by the author and is made available under the following Creative Commons Licence conditions.



CC creative commons
COMMONS DEED

Attribution-NonCommercial-NoDerivs 2.5

You are free:

- to copy, distribute, display, and perform the work

Under the following conditions:

BY: **Attribution.** You must attribute the work in the manner specified by the author or licensor.

Noncommercial. You may not use this work for commercial purposes.

No Derivative Works. You may not alter, transform, or build upon this work.

- For any reuse or distribution, you must make clear to others the license terms of this work.
- Any of these conditions can be waived if you get permission from the copyright holder.

Your fair use and other rights are in no way affected by the above.

This is a human-readable summary of the [Legal Code \(the full license\)](#).

[Disclaimer](#) 

For the full text of this licence, please go to:
<http://creativecommons.org/licenses/by-nc-nd/2.5/>

PERSONALISED FEEDBACK WITH SEMI-AUTOMATIC ASSESSMENT TOOL FOR CONCEPTUAL DATABASE MODEL

Firat Batmaz

Department of Computer Science
Loughborough University
Loughborough, LE11 3TU
f.batmaz@lboro.ac.uk

Roger Stone

Department of Computer Science
Loughborough University
Loughborough, LE11 3TU
r.g.stone@lboro.ac.uk

Chris Hinde

Department of Computer Science
Loughborough University
Loughborough, LE11 3TU
c.j.hinde@lboro.ac.uk

ABSTRACT

The increased presence of diagram-type student work in higher education has recently attracted researchers to look into the automation of diagram marking. This paper introduces web-based diagram drawing and marking tools for a new (semi-automatic) assessment approach. The approach reduces the number of diagram components marked by the human marker and provides individualised and detailed feedback to students. The tools which have been used in tutorials of a first year database module in the Computer Science department at Loughborough University are described together with findings from the usage of the tools.

Keywords

Conceptual Database Diagram, Diagram-Type Student Work, Improving Assessment and Student Feedback

1. INTRODUCTION

Feedback is an important part of formative assessment. It makes learners aware of any gaps that exist between their desired goal and their current knowledge, understanding, or skill [9]. Feedback on tests and homework is most useful when it provides specific suggestions for improvement [4]. Nicol and Macfarlane in their paper [8] presented a conceptual model of formative assessment & feedback and the seven principles of good feedback practice. Although feedback is widely accepted as a crucial part in the learning for assessment, in higher education, Gibbs and Simpson [5] argue that feedback to individual students in a class must have declined significantly as class sizes have increased.

Various systems have been developed to automatically mark student work and generate feedback. However there are very limited proven forms of automated assessment (e.g. multiple-choice, matching questions or simple 'fill in the blank' questions). These forms of automated assessment are not sophisticated enough to examine a student's understanding of complex content and thinking patterns [7]. Diagram-type student work (e.g. ER diagrams, UML) can be used to assess a student's knowledge in a more comprehensive way, as can essay-type student answers. Although there is some research into automatic essay and diagram marking (e.g. DEAP Project [11]), full automation of marking hasn't been achieved yet. We believe that human markers are still required to manually assess the diagrams in a supportive online environment. This environment can be seen as an intermediate stage for the future fully automatic systems.

We have developed a semi-automatic assessment system with the main characteristics of online diagram drawing and online marking of these diagrams by a human marker using a specialised tool. Based on this core system we have also developed an interactive environment that allows students to access detailed personalised feedback. Among hundreds of known diagrammatic notations in computer science, the research, in its initial stage, focuses on conceptual database diagrams (Entity Relationship diagrams).

The aim of the marking tool is to reduce the number of the components in the diagrams marked by the examiner, which is called semi-automatic marking in our research. That requires finding the identical components in different student diagrams. The Assess By Computer (ABC) [12] project, which has the same aim as this project, defines identical components by using those component's attributes (e.g. label, type,

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission.

© 2009 Higher Education Academy

Subject Centre for Information and Computer Sciences

adjacent boxes). In our research, identical components are defined by the references to the text describing the scenario (scenario referencing). A similar approach is used for the intelligent tutoring system in KERMIT [10] and ERM-VLE [10] projects. Scenario referencing and component matching are discussed in the Semi-automatic assessment section.

The KERMIT and ERM-VLE projects have developed their own online diagram editors to capture student diagrams. This research also requires its own diagram editor. The editor and marking tool are improved versions over those reported previously [6]. Discussion of the changes to the tools and the results obtained are presented below.

2. SEMI-AUTOMATIC ASSESSMENT

The semi-automatic marking aims to reduce the number of diagrams marked by the assessor. The system groups identical segments of the students' diagrams and then asks the assessor to approve the correctness of a diagram fragment from each of the different groups. Therefore the assessor would be involved in the marking process only for the number of diagram groups rather than the total number of student diagrams.

The correctness of the grouping depends on the criteria used to match the diagram pieces. The smallest diagram piece in each group can be either an entity or relationship component for a conceptual database diagram. This research uses the student design traces as criteria for each component in their diagram. Design traces are links from scenario texts to diagram components. So, for example, 'Lecturer' and 'HoD' might appear in a scenario and be initially mapped to corresponding entities which may at a later stage be merged into a 'Staff' entity. Thus there is a trace which links 'Lecturer' in the scenario to the entity 'Staff' in the diagram. Components of student diagrams for the same scenario are grouped based on these links in the semi-automatic approach. One component from each component group is represented with its design trace to the examiner. The examiners can see how the student created the components. It may enable the examiner to understand more and comment on the student's work.

The central part of the approach is the process of grouping diagram components. Too many component groups to be marked could decrease the efficiency of the assessment process severely. The number of groups depends on the number of different references for a particular component. The reason for different references is diversity in either the students' reasoning or their action for the same reasoning. The reasoning diversity is restricted by the given scenario text. The action diversity is limited by the user interface of the diagram editor. A method for scenario text writing was developed to control the reasoning diversity. The details of the method can be found in [2].

The marking part of the approach cannot change the number of component groups. However, some groups could be marked automatically if the ideal diagrams and previously marked components of a scenario are available. Automatic marking requires domain independent rules. The rules could be produced gradually after analysing the previously marked components for various scenarios. Each new rule generated will increase the efficiency of the system. Some example rules are explained in the marking tool section.

3. DIAGRAMMING TOOL

A special diagram editor was developed for the production of the design traces. The editor alters the traditional diagramming process in order to ease the trace production. The student doesn't draw a diagram for their design. They simply enter their component type and name and then the tool draws the student diagram. In this way, they can focus more on designing than drawing. The alteration enables students to follow the steps of the conceptual database design methodology taught in the database module.

The user interface of the editor is shown in Figure 1. It has four panes; Scenario, Tool bar, Diagram and General command. The Scenario pane shows the scenario paragraph by paragraph so that the student considers the information in that section only. This method is called "Scenario scaffolding" [3]. The scaffolding has a potential to reduce the action diversity. The Tool bar is used for creating components in the diagram pane. The Diagram pane automatically draws a diagram for a student design. General command panes are used for additional activities like submitting a diagram and displaying a diagram in UML notation.

Figure 1 shows an entity creation action as an example. To create a new entity type (or attribute) the student drags a noun-phrase directly from the scenario text in the current section and drops it onto the entity box in the tool bar and the new entity component is added to the diagram. In this way, a direct reference to the entity is captured. To create a relationship type, the student drags two noun-phrases and drops them onto the appropriate boxes in the bar. Then, in a pop-up dialog, the cardinality, optionality and name of the relationship are entered by the student. The direct reference of the relationship is deduced from the position of the two phrases in the scenario.

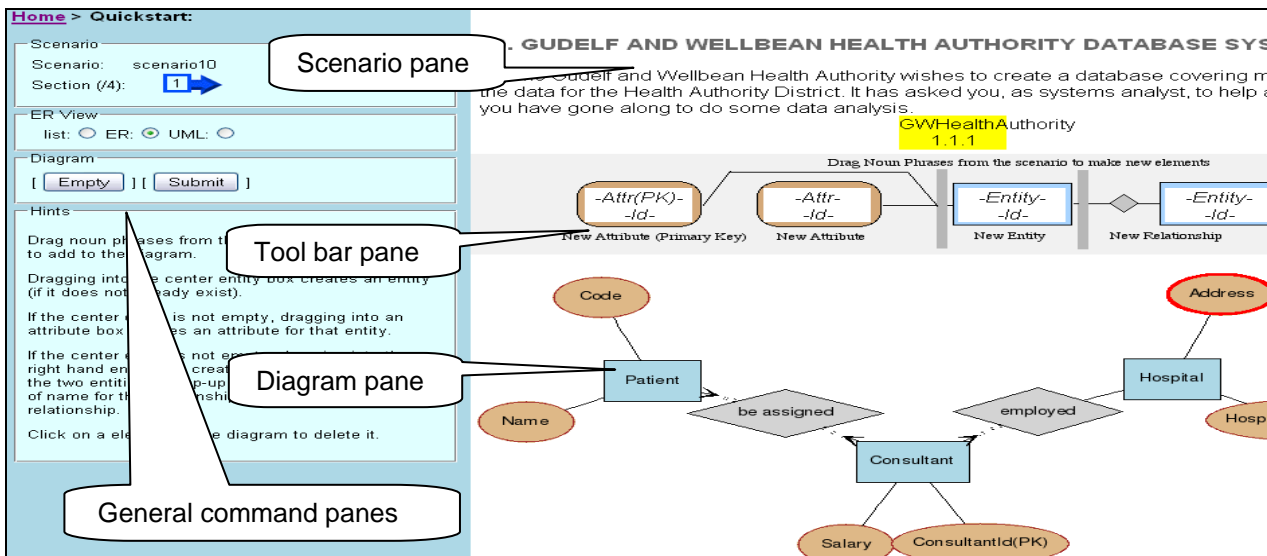


Figure 1 User interface of the diagramming tool

Scenarios used in the tutorials for this semester do not require creation of indirectly referenced components. The next version of the tool is already available which supports creating all types of components. However it has not been used in the tutorials yet.

4. MARKING TOOL

The marking tool is designed mainly to ease the partial marking style which enables semi-automatic assessment. It also needs to support the complete marking style which imitates paper-based marking. The details of complete marking can be found in [1]. Before the examiner marks diagram components partially, the engine part of the tool processes student diagrams in three phases; Normalisation, Grouping and Automatic marking.

Properties of diagram components used as matching criteria are put into a standard form in the normalisation phase to decrease the number of component groups. For example, Student S1 has got a relationship Rs1 between Entities E1 and E2 with a direct reference Ref1 which is represented as Rs1(E1,E2,Ref1). Student S2 has got a relationship Rs2(E2,E1,Ref1). After normalisation, Rs2 is written in Rs2(E1,E2,Ref1) so that Rs1 and Rs2 are in the same group. The second phase is to group components by using matching criteria. The criteria of a direct referenced entity is a noun phrase reference (e.g. Es1 (Ref2)). For each attribute, the criteria are their noun phrase reference and related entity (e.g. As1(E1,Ref3)). For relationships a sentence reference is used with the participating entities (e.g. Rs1(E1,E2,Ref1)).

Third phase marks some components automatically. The basic rule is to use the examiner's ideal diagram. If one of the group components belongs to the ideal diagram, all the other members become correct (e.g. Gcorrect(Rs1,Rs2,Rs3,Rteacher)). If many students get components correct then the automation will be increased. Another rule is to use previously marked components. If an entity is wrong, all attribute and relationship components related to that entity become wrong (e.g. As1wrong(E1wrong,Ref3) or Rs1wrong(E1wrong,E2,Ref1)). If the wrong entity is a common mistake, many diagram components are marked automatically and this increases the efficiency of the system. Figure 2a shows the command pane of

the editor. "Prepare" command does the normalisation and grouping. "Mark by" and "Pre-Rel" command do the automatic marking.

Figure 2b shows the interface of the partial marking editor. The tool displays the unmarked components to the examiner. The examiner first sees all entity components, then attribute and finally relationship components. If the component is an entity type then only its noun phrase reference is displayed. If it is an attribute then its noun phrase reference and entity is shown. If it is a relationship, then its reference sentence and participant entities are presented. Components are marked as correct, accept or incorrect. The examiner could also add a comment to each mark or use a standard one during marking.

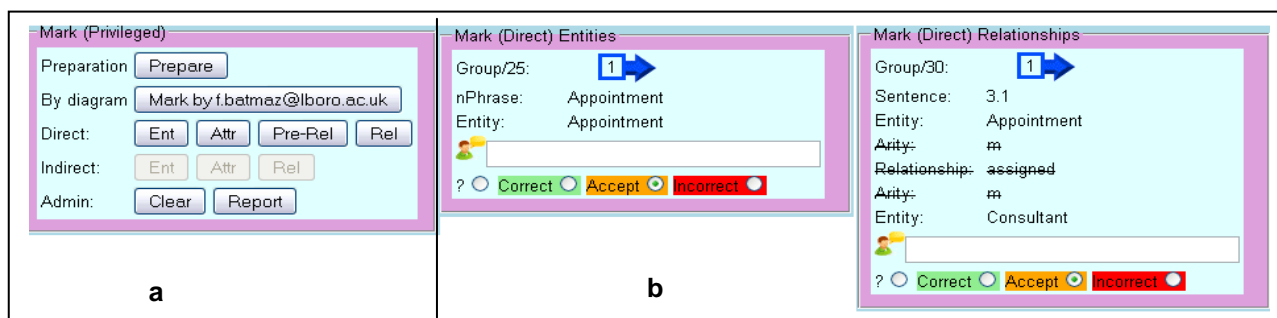


Figure 2 Screenshots of part of the editor interface

The student diagrams are released after the marking is complete. For the student this means that they see another option (checkbox) on their screens which allows them to view the marked version of their diagram. Using this they can view their diagram now coloured in a combination of red, amber and green. If the student moves the mouse pointer over any component in their diagrams, they can see the specific comment for it. In this way the student gets detailed personalised feedback for their work.

5. EXPERIMENT AND RESULTS

The diagram editor has been used with two scenarios by a first year class of 200 students. This represents 4 separate sessions since the first year class had to be split into four groups of 50 students. At the beginning of each 50min practical session a short demonstration of the system was given showing how to make one entity, one attribute and one relationship. The students were then asked to create ER diagrams for the first scenario (hitherto unseen) and if possible go on to do the same for the second scenario. By the end of the session most students had finished the first scenario and some had finished both.

Scenario No	Total Component #	Component Group #	Diversity Rate %	Auto marked Group #	Manual marked Group #	Efficiency Rate %
1	5356	708	13	468	240	96
2	3707	607	16	317	290	92

Table 1 Summary of diagram marking for two scenarios

The tutor marked the student diagrams by the marking tool in less than two hours. Table 1 shows the summary of the marking task. The tutor marked 240 components out of 5356. That makes the efficiency of the tool for first scenario approximately 96 percent. The efficiency for the second scenario is 92 percent. This might be interpreted such that the system's efficiency increases when more students draw ER diagrams for the same scenario. The table also shows the diversity rate. This rate is calculated by using both component group and total component numbers. If student diagrams are similar to each other then the diversity rate decreases otherwise it increases. Diversity rate could be used as a feedback for the scenarios. Student solutions for the similar scenarios should have the same rate. If they are very different, the tutor may analyse revise the scenario text to find out the reason for the diversity.

The marking tool produces a detailed report for any chosen scenario. The report has a list of every distinct element, how it was marked and the number of students whose diagrams included that element. This reveals for example how many students made the 'same mistake'. So for example it was clear that something in the way the first scenario was worded caused 50 students to wrongly identify "Consultant Name" as an entity and go on to make related mistakes with attributes. The report contains aggregate marks e.g. entities (green 82%,

amber 7%, red 11%) attributes (green 82%, amber 10%, red 8%) and relationships (green 34%, amber 59%, red 7%) showing that it is the precise identification of relationships (amber 59%) that caused the most problems.

In feedback sessions, the students were able to see their marked diagrams. The colour coding of tutor comments was extremely well received by the students and led to lively, positive discussion of the principles involved with interpreting the scenarios which was very beneficial. A simple questionnaire about the editor and the associated marking feedback has been given to students at end of the term. The results from 70 returns show that the students were favourably disposed to the editor and they liked the coloured feedback.

6. CONCLUSION AND FURTHER WORKS

Students have readily accepted the concept of using the editor as a way of submitting their work to scrutiny by the tutor in return for more extensive and personalised feedback than would have been possible otherwise. The marking tool has decreased the workload of the tutor and gives the feedback about students' diagramming skills and scenario used in the assessment. Now that a basic system has been implemented where the components of the diagram are directly referenced to the scenario, the next stage is to allow the user to create indirect referenced components by splitting and merging existing elements. The marking tool will be improved to support marking of these components.

7. ACKNOWLEDGEMENT

Development of the public version of the semi-automatic system is in receipt of general development fund from the Higher Education Academy subject network for Information and Computer Sciences. Their support is gratefully acknowledged.

8. REFERENCES

- [1] Batmaz, F. and Hinde, C.J., 2007, A Web-Based Semi-Automatic Assessment Tool for Conceptual Database Diagram, Proceedings of the sixth Web-Based Education conference, pp. 427-432.
- [2] Batmaz, F. and Hinde, C.J., 2008, A Method For Controlling The Scenario Writing For The Assessment Of Conceptual Database Model, Proceedings of Computers and Advanced Technology in Education.
- [3] Conati, C. and VanLehn, K., Toward computer-based support of meta-cognitive skills: a computational framework to coach self-explanation. *International Journal of Artificial Intelligence in Education* 11, 2000, 398-415.
- [4] Elawar, M.C., and Corno, L. 1985 A factorial experiment in teachers' written feedback on student homework: Changing teacher behaviour a little rather than a lot. *Journal of Educational Psychology*, 77 (2): 162-173.
- [5] Gibbs, G. and Simpson, C. Conditions under which assessment supports students' learning. *Learning and Teaching in Higher Education*, 1, 2004-05.
- [6] Hall, L. and Gordon, A., 1998, A virtual learning environment for entity relationship modelling. Proc 29th SIGCSE, Atlanta, Georgia, US, 1998, 345-349.
- [7] Heinrich, E. and Wang, Y., Online marking of essay-type assignments. Proc. ED-MEDIA 2003, Honolulu, Hawaii, USA, 2003, 768-772.
- [8] Nicol, D. & Macfarlane-Dick, D. 2004 Rethinking Formative Assessment in HE: a theoretical model and seven principles of good feedback practice, http://www.heacademy.ac.uk/assessment/ASS051D_SENLEF_model.doc
- [9] Sadler, D.R. 1989 Formative assessment and the design of instructional systems. *Instructional Science*, 18 (2): 119-144.
- [10] Suraweera, P., and Mitrovic, A., 2002, KERMIT: A constraint-based tutor for database modelling. Proc ITS'2002, LCNS 2363, 2002, 377-387.
- [11] Thomas, P., Waugh, K. and Smith, N., Experiments in the automatic marking of ER-diagrams, *SIGCSE Bull.* 37, 3 (Sep. 2005), 158-162.
- [12] Tselonis, C., Sargeant, J. and McGee Wood, M., Diagram matching for HCC assessment, Proc. 9th CAA Conference, Loughborough UK, 2005