Automatically assessing graph-based diagrams

P.G. Thomas, N. Smith, K. Waugh

Centre for Research in Computing

The Open University

Walton Hall

Milton Keynes

MK7 6AA

{P.G.Thomas, N.Smith, K.G.Waugh}@open.ac.uk

Abstract

To date there has been very little work on the machine understanding of imprecise diagrams, such as diagrams drawn by students in response to assessment questions. Imprecise diagrams exhibit faults such as missing, extraneous, and incorrectly formed elements. The semantics of imprecise diagrams are difficult to determine. While there have successful attempts at assessing text (essays) automatically, little success with diagrams has been reported. In this paper, we explain an approach to the automatic interpretation of graph-based diagrams based on a 5-stage framework. The paper describes our approach to automatically grading graph-based diagrams and reports on some experiments into the automatic grading of student diagrams. The diagrams were produced under examination conditions and the output of the automatic marker was compared with the original human marks across a large number of diagrams. The experiments show good agreement between the performance of the automatic marker and the human markers. The paper also describes how the automatic marking algorithm has been incorporated into a variety of software teaching and learning tools. One tool supports the human grading of entity-relationship diagrams. Another tool is for student use during the revision of entity-relationship diagrams. This tool automatically marks student answers in real-time and provides dynamically-created feedback to help guide the student's progress.

1. Introduction

Diagrams are an extremely useful mechanism for summarising information and displaying relationships between objects (Anderson, Meyer and Olivier 2002; Marriott and Meyer 1998). They are used in education to explain concepts and elicit understanding. While there are many software tools for supporting the drawing of diagrams (e.g. Rational Rose and Violet), the automatic understanding of such diagrams is a difficult problem. Some progress has been made on automatic understanding when diagrams conform rigidly to a diagrammatic grammar (Marriott, Meyer and Wittenburg. 1998; Flower, Masthoff and Stapleton 2004). However, the problem of general diagram understanding is wider than this. In teaching, student-drawn diagrams can be partially incorrect yet still have merit. Trying to extract useful information from such diagrams automatically is even more difficult. We refer to such diagrams as *imprecise* by which we mean that elements can be missing, there can be extraneous elements, and elements that are constructed incorrectly (Smith, Thomas and Waugh 2004). Therefore, analysing a student diagram to see whether it contains appropriate and correct information is a challenge. If it were possible to analyse a student produced diagram automatically it would be possible to provide feedback to the student on those parts of the diagram that were found to be in error.

In this paper we shall describe both our general approach to analysing imprecise diagrams and how we have used our technology in particular in a set of software tools designed for learning and assessing graph-based diagrams. Typical examples of graph-based diagrams are entity-relationship diagrams (ERDs), Unified Modelling Language (UML) diagrams, biological flow diagrams and chemical structure diagrams. In the work described here, we have used entity-relationship diagrams (ERDs) as an exemplar of graph-based diagrams.

In a teaching environment, we would prefer to have tools that are less constraining than those used by professional modellers which allow students to express their understanding – or lack of it – and have the tools give feedback on the errors being made. That is, the tools would deal with imprecise diagrams. Such tools would enable a student to investigate a problem in a variety of ways. Tools that provide this kind of feedback could also be used to grade student work and one can imagine such tools being an integral part of a computer aided assessment system.

Our interest in assessing student attempts at drawing ERDs stemmed initially from our attempts to mark (grade) online examinations, albeit in a formative environment (Thomas et al., 2002, Thomas, 2003, 2004). While there are several systems being developed for grading textual material (Burnstein Chodorow and Leacock 2003; Shermis and Burstein 2003; Haley et al., 2005) and there is a considerable literature for describing diagrams (see, for example, papers by Anderson and McCartney (2003), Chock and Marriott (1995), Kniverton (1996) and Marriott, Meyer and Wittenburg (1998)) there is very little work on grading diagrams. Tsintsfas (2002) has produced a framework for the assessment of diagram-based coursework which has fed into an ERD tool within the CourseMarker CBA system. Higgins and Bligh (2006) and Batmaz and Hinde (2006, 2007) have investigated semi-automatic marking systems. We decided to investigate the extent to which it would be possible to produce an effective, fully-automatic marking tool.

There are two significant elements to successful automatic grading of a free-form diagram (a diagram in which the user is essentially unconstrained in what they draw). We require a method for identifying the correct elements in a diagram and a method for applying a marking scheme to the diagram. To test the effectiveness of these elements, corpora of accurately marked diagrams produced in realistic situations are needed in order to compare an automatic marker's performance against that of human markers. The questions that we shall address in this paper are how accurately our approach marks ERDs and whether the approach is suitable for formative purposes.

In the following sections we discuss the general problem of diagram interpretation, our particular approach to automatically marking a diagram, and the tools we have built to exploit this technology for learning and assessment including the feedback that can be provided to students. The final section discusses how we intend to take this work forward.

2. Diagram interpretation

Figure 1 shows a typical ERD. It is feature-based in the sense that it is composed of several types of features: boxes with names inside, lines with names beside them, circles (either open or filled) and, at the ends of some lines, 'crowsfeet'. ERDs are typical of graph-based diagrams in which nodes (usually denoted by circles or rectangles) are connected by edges (usually denoted by lines) and both nodes and edges have attributes (denoted by adornments such as text, circles and crowsfeet). In use, nodes represent objects of some kind and edges represent associations between the nodes. Precisely what an association means varies from one type of diagram to another. In much of what follows, the precise meaning of a diagram, and the fact that it is an entity-relationship diagram, is not significant: it is the structure of the diagram that we shall be concentrating upon.

Each of the features in a diagram has meaning within the context of the diagram. For example, in an ERD a box represents an entity type in which the name serves to label the type of entity being described. A line with a crowsfoot at one end represents a one-to-many relationship between the entity types at either end of the line. However, some features, if they were to appear *on their own* would not convey meaning. An open circle on its own carries no meaning in the context of an ERD, but if it appears at the end of a line joining two entity types, it represents an optional participation.

It turns out to be useful to identify diagram features (or small combinations of features) that can appear on their own and carry meaning. In an ERD, there are three such structures: an individual named box represents an entity type, a named line joining two entity types represents a relationship in which the name labels the type of relationship, and a box wholly containing another box represents a supertype-subtype association (examples can be found in Figure 2). We refer to instances of these three structures as minimal meaningful units (MMUs) because, if any feature of the structure is removed, it ceases to convey meaning in this domain. Generally speaking, MMUs can be aggregated to form larger structures with more meaning. But since these diagrams are no longer minimal they are called meaningful units (MUs). Thus, the diagram in Figure 1 is an MU, and it contains 6 entity type MMUs and 7 relationship MMUs. The diagram in Figure 2 contains 8 entities, 4 supertype-subtype associations and 5 relationships. In other types of graph-based diagrams there can be more than three types of MMU. In our approach to diagram marking, we look for instances of MMUs and then, where appropriate, combine (aggregate) the MMUs into MUs and award marks for correct MUs.

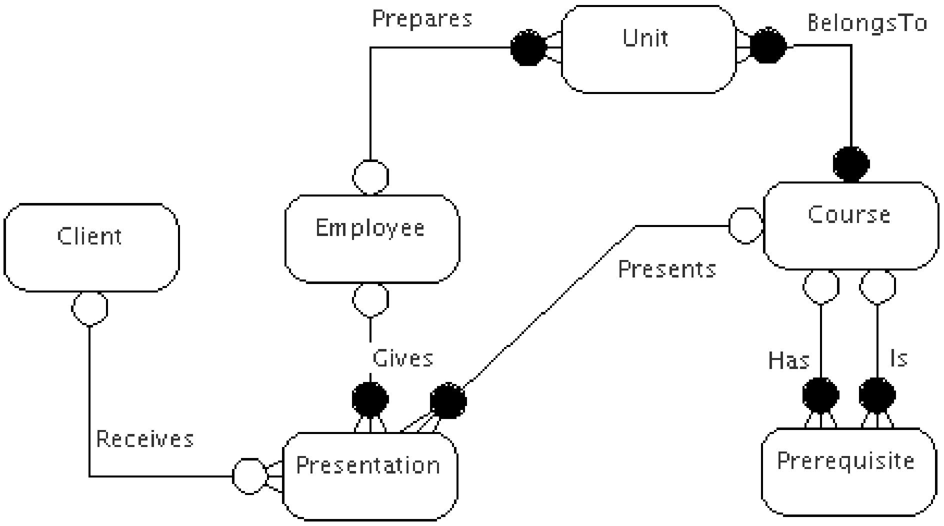


Figure 1 An entity-relationship diagram (ERD)

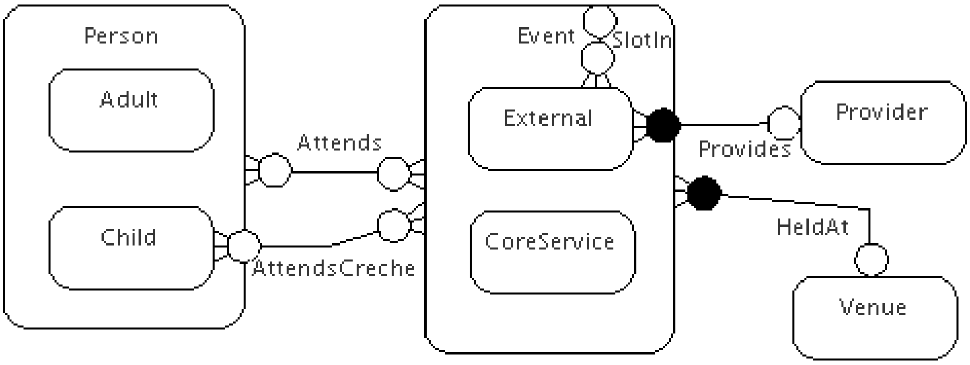


Figure 2 An ERD containing supertype-subtype associations

In the domain of data modelling, there are certain small diagrams (MUs) that have useful properties. For example, Figure 3 illustrates an equivalence between two MUs. The MU at the bottom consists of two one-to-many relationships with specific participations. This aggregate MU can be interpreted – in this domain – as equivalent to the MU (actually an MMU) shown above it. From a grading perspective, it is essential to be able to recognise such equivalences as both must be marked equivalently.

These observations, among others, have led us to the five stage framework for interpreting imprecise diagrams shown in Figure 4. This is a framework for interpreting diagrams *automatically*, and is not meant to simulate the process that a human marker might adopt in understanding a diagram.

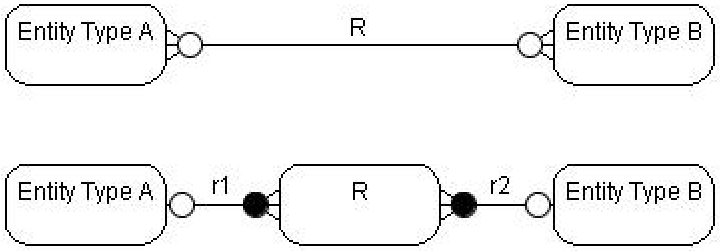


Figure 3 Equivalent MUs

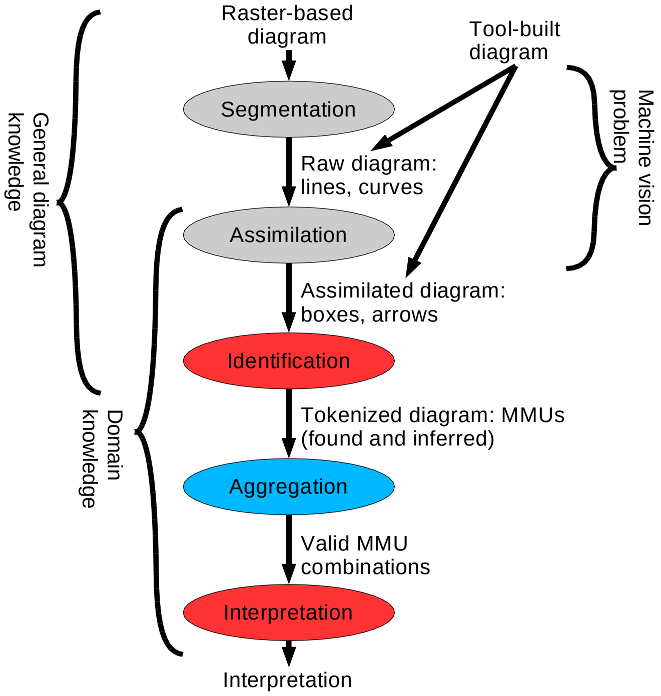


Figure 4 The 5-stage framework for automatic imprecise diagram interpretation

The framework consists of five stages (the flow is from top to bottom in Figure 4). The first two stages of segmentation and assimilation) are concerned with turning an on-screen diagram (raster-based) into a representation in which features such as boxes, lines and arrows have been recognised. Much work has been done in this area (see, for example, Hammond & Davis (2005)). Our primary interest is in the last three stages where domain specific knowledge is required to interpret the diagram. The identification stage identifies MMUs. The aggregation phase combines MMUs to form useful MUs, and the final stage takes the collection of MUs and interprets their meaning. In the work reported here, we say that we have successfully interpreted a diagram when we have successfully graded the diagram and provided meaningful feedback.

3. Automatic marking

In this section we shall describe the latter three stages of our framework. Diagrams are drawn using a drawing tool. This tool provides only limited functionality in that it allows the user to draw the features of an ERD but provides little support for drawing correct diagrams as would be found in a professional drawing tool. The output from this tool is the output one would expect from the second stage of the framework and feeds directly into the third, identification, stage.

We shall illustrate our approach by taking the example shown in Figure 5 which shows an ERD which was drawn by a student in answer to an examination question to which Figure 1 was the expected solution (the model answer). The aim is to compare the two diagrams, determine their similarities and award marks appropriately.

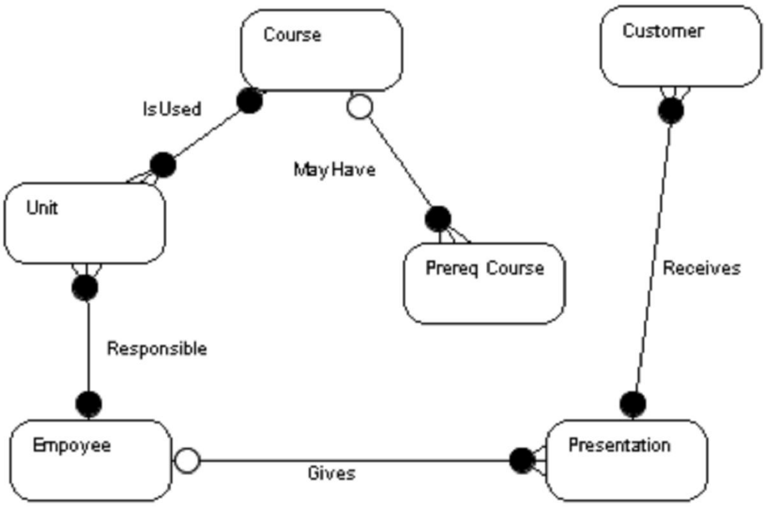


Figure 5 A student drawn ERD

3.1. Identification

The input to the identification stage is a collection of drawing features (texts, boxes, lines circles and crowsfeet) together with geometric data such as location on the drawing canvas. The identification stage transforms this data into a collection of MMUs. It recognises, for example, that the line labelled ‘Receives’ in Figure 5 represents a one-to-many relationship between the entities named ‘Presentation’ and ‘Customer’ and that the relationship has mandatory participations at both ends. The output from the identification stage consists of the MMUs found in the diagram categorised by type of MMU. This data is the input the aggregation stage which we shall describe once the approach to interpretation has been discussed.

3.2. Interpretation

The interpretation stage compares the student answer diagram with the model answer diagram by comparing the sets of MMUs (together with MUs found in the aggregation stage) in the two diagrams. Thus, the entity types in the student diagram are compared with the entity types in the model answer and, likewise, the two sets of relationships are compared. The aim of the comparisons is to determine the best possible match between the (M)MUs in the student answer diagram and the same types of (M)MU in the model answer.

A comparison is performed using a similarity measure (a real number in the range [0..1] with 1 representing an exact match). That is, each (M)MU of a given type in one diagram is compared with every (M)MU of the same type in the other diagram and each time a measure of their similarity is computed. The best possible match between two diagrams is found by maximising the sum of the similarities for each (M)MU. Matches with similarity values below a specified threshold are discarded on the grounds that the level of similarity is too low for the pair of (M)MUs to be considered even partially the same. How the value of the threshold and other parameters are determined is discussed later.

When the automatic marker identifies several possible alignments that could be made between the student diagram and the model answer (or several model answers in the case of alternative solutions), it determines which matching is the most plausible (i.e. with the highest overall similarity) and awards marks appropriately.

The advantage of computing a numeric similarity measure is that it enables the graceful handling of imprecision in a student's diagram. For example, when comparing the names of entity types and relationships, we use edit distance between the names as the basis of the similarity measure. Other similarity measures can be determined for the adornments (crowsfeet and blobs) on any relationship line and these measures can be combined into a single similarity measure for each MMU. The precise manner in which these different similarity measures are combined is defined by the marking scheme (see section 3.5 below).

3.3. Text matching

In ERDs, much of the information used to determine matches between the MMUs in the two diagrams comes from the names given to entities and relationships. As mentioned above, edit distance provides a simple measure of similarity between names. This simple scheme has the advantage that simple spelling errors can be taken into account (see the misspelling of 'Employee' in Figure 5: 'Empoyee' is very similar to 'Employee'). However, the simple scheme really only works well when the student uses simple names that are close to those used in the model answer. Dealing with more complex naming requires a number of issues to be handled effectively. Using ideas from natural language processing (NLP), we treat entity names as noun phrases and relationship names as verb phrases and base comparisons on the head words of the phrases, after stemming to remove inflectional endings (such as '-ed' and '-ing'). Function words in the phrases are discarded. The paper by Thomas, Waugh, and Smith (2007c) describes this in more detail.

3.4. Synonyms

Of particular importance in accurate automatic marking is the detection of synonyms. Synonyms in everyday speech can be dealt with by conventional look-up techniques, and domain specific synonyms can be handled similarly. Other names can be used synonymously. For example, 'Prereq Course' is a hyponym. If the model answer used the word 'Course', then 'Prereq Course' can be assumed to be a synonym because it represents a type of 'Course'. However, in Figure 5 this would mean that 'Prereq Course' would be considered to be similar to both 'Course' and 'Prerequisite', which appear in the model answer in Figure 1. This is not necessarily a problem provided that the similarities of 'Prereq Course' and 'Prerequisite' to 'Course' are less than 1 to allow the similarity of the occurrence of 'Course' in both diagrams to be bigger, at precisely 1. Since the aim is to determine the best overall match between the MMUs of the two diagrams then, provided the matching of words that might possibly be similar does not interfere with the matching of truly synonymous words, the scheme works well.

The automatic marker begins by finding close matches between entity names in the two diagrams taking known synonyms into account. If there are any entities in the student answer and the model answer not matched by this process, their 'contexts' (the immediate relationships and supertype associations that the entities participate in) are compared. If the two contexts match sufficiently well, the two entities are considered a match.

Such rules have to be applied carefully. In diagrams where supertype-subtype associations are present, hyponymy is to be expected and it is necessary to match supertype with supertype and subtype with subtype and not allow the obvious similarity between supertype and subtype names to confuse the matching. We solved this problem by a scheme in which the internal representation of a name (as a noun or verb phrase) is appended with the name's full supertype hierarchy before their similarity is computed. For example, 'Adult' becomes 'Adult_Person' and 'External' becomes External_Event'.

The similarity measures incorporate a number of weights and thresholds. The weights are used to change the significance of diagram features to reflect their importance in different domains. The thresholds are used to avoid matching MMUs where the evidence (similarity) is limited. In some cases, the level of a threshold has been determined by experiment.

3.5. Aggregation

We use aggregation to account for equivalences in ERDs. For example, a many-to-many relationship may be replaced by an equivalent pair of one-to-many relationships with a new entity type in common (as in Figure 3). Occasionally students attempt to apply this technique (although not always correctly). This means that a student's diagram may contain an entity type participating in (at least) two one-to-many relationships that do not appear in the model answer. Our automatic marker will examine any unmatched parts of a student's diagram to see whether it might be the result of an attempt to expand a many-to-many relationship. If so, the marker will aggregate the pair of one-to-many relationships and their common entity type into a single many-to-many relationship and re-grade the resulting diagram.

Given that the application of an equivalence may be performed, we have to accept evidence that an attempt to make a transformation has been made which is weaker than looking for a correctly formulated pair of one-to-many relationships. Our approach throughout this work is always to try to apply processing steps that will improve the grade for a diagram. Therefore, if an equivalence is suspected, the aggregation step is performed which, if it does not improve the grade does no harm because the step is only performed on those parts of a diagram which do not currently contribute to the grade.

3.6. Marking schemes

Having found the best possible match between sets of MMUs, a marking scheme can be applied. In general, there can be a wide variety of marking schemes, often depending upon the concepts being assessed at a particular stage of a course. This makes marking schemes quite difficult to construct for automatic processing purposes. One approach would be to provide a scripting language for specifying marking schemes but it is doubtful whether it would be used in practice. Therefore, we devised a parameterised marking algorithm which, while not comprehensive, is adequate for marking many uses of ERDs in different learning contexts.

The automatic marker awards marks for each feature it identifies in the student diagram giving a proportion of the mark for a partially correct feature (based on the similarity measure it computed). The user can set a threshold on the award of proportional marks below which no marks are awarded. It is also possible to limit the number of marks awarded to the set of instances of an MMU (to allow marking schemes of the form, 'award one mark for each correctly identified entity type up to a maximum of 5 marks'). The marker can be instructed to mark to the nearest half or whole mark.

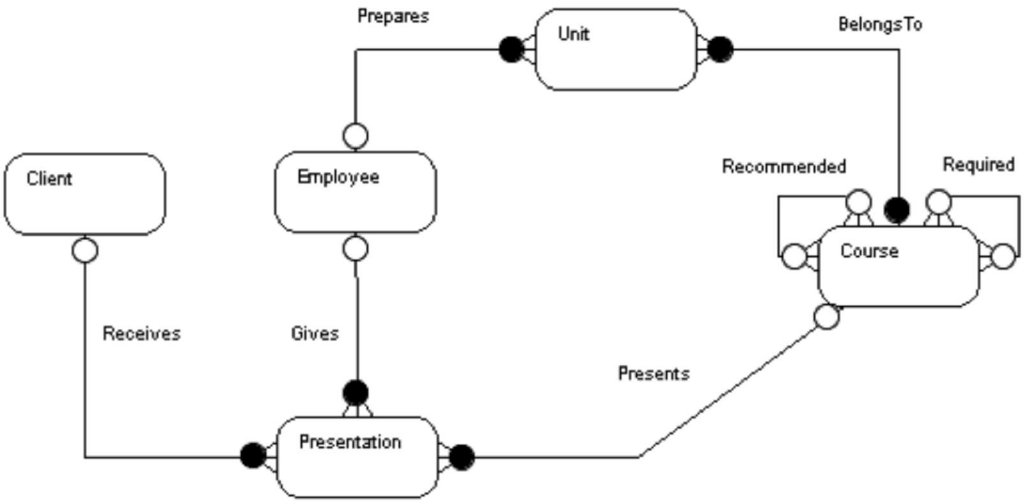


Figure 6 The alternative recursive relationships in the second solution

Alternative solutions to a question are handled by having more than one marking scheme for that question, one for each alternative solution. The automatic marker marks the student diagram according to each possible marking scheme and awards the mark found for the most plausible of the alternative solutions. For example, Figure 6 shows an alternative solution to the ERD shown in Figure 1.

4. Experimentation and results

In our early experiments with marking ERDs (Waugh, Thomas and Smith 2004; Thomas, Waugh and Smith 2005) we gained experience with relatively small sets of diagrams. These experiments gave us confidence in the approach but we required a much larger corpus of diagrams to provide more convincing evidence. Therefore, we set about constructing a sizeable corpus of student drawn ERDs. For our first, larger corpus we gathered 591 diagrams drawn in a real, invigilated examination (the diagram shown in Figure 1 is the model answer to the question posed). This corpus only contains two of the three types of MMU found in ERDs, but does have two alternative solutions (Figures 1 and 6). Preliminary results with this corpus were reported elsewhere (Thomas, Waugh and Smith 2005, 2007b, 2007c). Our second corpus is smaller (169 diagrams) but was used to ensure that the marker used all three types of MMU; Figure 2 shows the model answer for this question.

We collected three marks for each diagram in both corpora. The first mark was the one awarded by the exam marker, an experienced tutor. There were several such markers used on both corpora. The second mark was produced independently by a member of the course team who was not one of the exam markers. All markers' work was reviewed by another course team member to ensure reliability. As the second marks were all produced by the same person, we consider them to be more consistent than the first marks; comparing these marks allows us to investigate the reliability of markers. The third mark was the computer generated mark.

We adopted the reviewed marks as the 'gold standard' of marking – the 'true' marks deserved by the scripts. We can judge the reliability of the marking process (human and automatic) by comparing how the first set of human marks and the automatic marks vary from the 'true' marks. Given that the first set of marks are considered accurate enough (they were accepted for examination purposes), the divergence in marking between the first and second human marks is an acceptable threshold. If we find that variation between the automatic marks and the true marks is no more than this, the automatic marker performs at least as well as a human marker. Table 1 shows the differences between the two sets of human marks.

Table 1 Differences between the two sets of human marks

Difference		**0**	**0.5**	**1**	**1.5**	**2**	**2.5**	**3**	**3.5**	**4**
First corpus	**Number**	302	177	74	28	6	2	1	0	1
	%	51.1	29.95	12.52	4.74	1.06	0.34	0.17	0	0.17
Second corpus	**Number**	50	68	35	12	1	3	0	0	0
	%	29.59	40.24	20.71	7.1	0.59	1.78	0	0	0

The striking feature of this data is the high number of diagrams that differed in their mark: almost half the diagrams (49%) in the first corpus and more than two-thirds (70%) in the second corpus, although the change was only a half mark in 30% of cases in the first corpus and 40% in the second corpus. Nevertheless, this represents an example of the known variability in human marking (Newstead & Dennis, 1994). This does not mean that all changes were due to marking error: there were some transcription errors and, it can be argued, some are cases where the markers genuinely do not agree, due to subjectivity in the marker. In practice, this degree of variation in the marks was sufficiently small that either set of marks would have been acceptable for assessment purposes. If the automatic marker is found to agree at least as well as this with the 'true' marks, we can judge the automatic marking to be equally acceptable.

Two measures were used to determine the agreement between markers: Fleiss's generalised kappa measure (Fleiss 1971) and Gwet's AC1 measure (Gwet 2001). We consider Gwet's measure to be superior, as it more accurately accounts for chance agreement between markers (see Gwet (2001) for details). We present the kappa measure to enable easier comparison with other marking approaches. Critical values for these measures are around 0.15 for both AC1 and kappa: agreement measures above these values allow us to reject, with over 99% confidence, the null hypothesis that the marks are allocated randomly.

The automatic marker has a number of parameters (weights and thresholds) that can be tuned for a specific data set. Applying the marker to the training set allowed us to set the values of these parameters to maximise its performance, defined by the agreement measures. For example, the value of the threshold which determines whether there is sufficient agreement between individual elements of two diagrams for a mark to be awarded was set at its optimal value for each of the testing sets: 70% for the first set but 60% for the second.

We arbitrarily divided each corpus into two sets of diagrams. The first, training, set (197 diagrams for the first corpus and 72 diagrams from the second) was used for setting the parameters that control the marking behaviour. The remaining diagrams (394 for the first corpus, 97 from the second) formed the testing set and were used to test the accuracy of the automatic marker when compared to the true human marks. No parameters or thresholds were altered during the testing of the marker.

4.1. First corpus experiments

After training the marker on the first corpus' training set, we applied the marker to the 394 diagrams in the testing set. Table 2 shows the results of this experiment. The diagrams were marked out of 7 and rounded to the nearest half mark.

Table 2 Differences between the automatic marker and true marks (first corpus)

Difference	0	0.5	1	1.5	2	2.5
Number	266	93	32	3	0	0
%	67.51	23.6	8.12	0.76	0	0
Cum. %	67.51	91.11	99.23	100	100	100

The results show that the automatic marker agrees exactly with the true human marks in over 67% of cases. However, given the variation often present in human marking we feel that the percentage of cases which differ by no more than half a mark is a better reflection of the automatic marker's accuracy: approximately 91% are in this category.

In the worst cases (where the auto mark differed from the human mark by 2 or more) the difference can be accounted for primarily by the students' use of recursive relationships in place of binary relationships. Two of the worst cases contain an error in which the students had named two entity types identically; the human markers had given 'the benefit of the doubt' and awarded credit for something the automatic marker had viewed as a significant error. Which is the better approach is a matter for debate. In two other cases the automatic marker failed to recognise the student devised entity names 'Course Requirements' and 'Pre Req Course' as synonyms for the model answer's 'Prerequisite'.

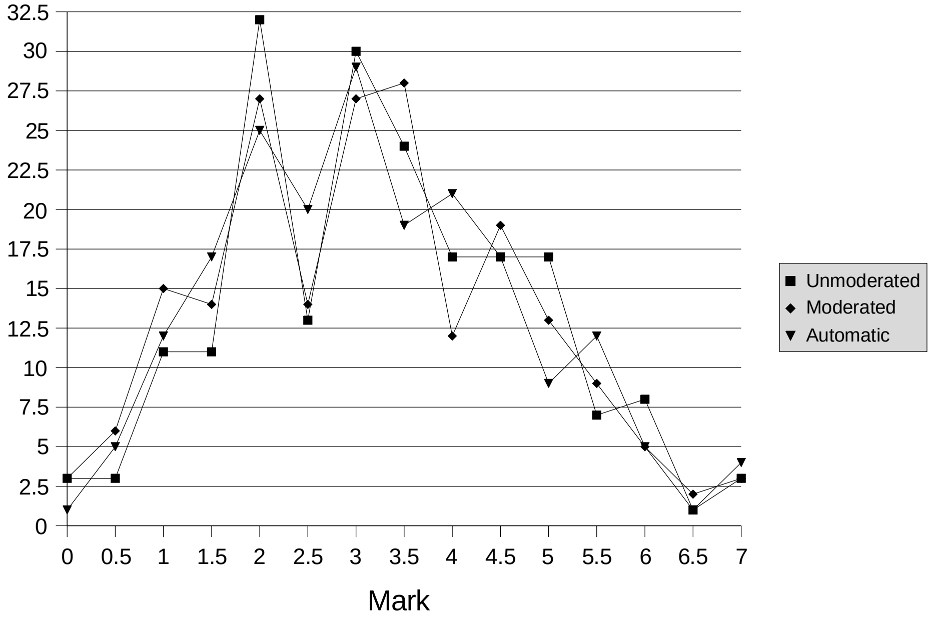


Figure 7 Comparison of marks across the range (first corpus)

Figure 7 shows how well the automatic marker agrees with the human markers across the range of marks (moderated = true human marks; unmoderated = first human marks). Compared with earlier experiments on the automatic marking of textual answers (Thomas, Waugh and Smith 2005), in which the automatic marker failed at both ends of the range, the diagram marker fits very well. The difference between the automatic marker and the human markers was on average 0.21 with a standard deviation of 0.34.

Table 3: Inter-marker agreement for the first corpus

(First = the first marks, True = the 'true' marks', Auto = the computer generated marks)

	True - First		True - Auto		First - Auto	
	AC1	Kappa	AC1	Kappa	AC1	Kappa
Training set, half marks	0.4794	0.4605	0.6311	0.6194	0.3926	0.3715
Training set, whole marks	0.5927	0.5636	0.7589	0.7417	0.5525	0.5181
Testing set, half marks	0.2249	0.2335	0.3064	0.3144	0.2039	0.2127
Testing set, first half, half marks	0.4682	0.4518	0.6363	0.6259	0.4248	0.4518
Testing set, second half, half marks	0.4789	0.4657	0.6634	0.6538	0.4356	0.4206

The AC1 and kappa measures (Table 3) show a high degree of conformity between the markers. In every case, the agreement measures are well above 0.15 and therefore show that we can reject the null hypothesis that the marks were awarded randomly. The first column of data in Table 3 shows the degree of agreement between the human markers. This level of disagreement between markers is considered acceptable. If the automatic marker can produce marks that agree at least this strongly with the human markers, the automatic marker is working acceptably well. The second and third columns of Table 3 show how well the automatic marker agrees with the true and first marks respectively. In particular, if the numbers in the third column are as high as the numbers in the second, the automatic marker is as accurate as a human marker.

The agreement measures do not take account of the degree of disagreement between markers (a difference of half a mark is counted as strongly as a difference of seven marks). The second row of Table 3 shows the agreement measures on the testing corpus, but with all marks rounded to the nearest whole mark. As can be seen, the effect on the agreement measures is not large for this corpus.

The remainder of Table 3 shows the agreement measures between the first set of marks, the 'true' marks, and the automatically-produced marks for the testing data set. The third result row of Table 3 shows the measures for the whole of the testing set. As the agreement measures naturally fall as the size of the sample increases, we arbitrarily split the testing set into two parts, each of which is about the size of the training set; the agreement measures for these two sets are shown in the last two rows of Table 3 and can be directly compared to the agreement measures for the training set. As can be seen, the performance on the training and testing sets is very close, showing that there was no over-fitting to the training data.

All these results tell the same story: the automatic marker agrees with the 'true' marks far more strongly than the first set of human marks agree with the 'true' human marks. As we consider the 'true' marks to be the 'gold standard', the most accurate marks for the diagrams, we can say that the automatic marker marks better than most human markers. The cost of this agreement is that agreement between the automatic marker and the first set of marks is less than the agreement between the first and 'true' marks.

Finally, when we look at whether the automatic marker awards consistently higher or lower marks than the humans we find that there is little bias. Of the 128 diagrams where the two did not agree precisely, the automatic marker awarded more marks than the humans in 72 cases (56%) and fewer marks in 56 cases (44%).

We conclude, therefore, that for the first corpus the performance of the automatic marker provides an accurate result (i.e. within 0.5 of a mark) in 91% of cases with only 3 cases (out of 394) where the difference was 1.5 giving any cause for concern. The agreement measures show that the automatic marker is more accurate than the human marker.

4.2. Second corpus experiments

The second corpus consisted of 169 diagrams taken from answers to an examination question in which all three ERD MMUs were expected. After training the marker on a training set of 72 diagrams, we applied the marker to the remaining 97 diagrams. Table 4 shows the results of this experiment. Again, the diagrams were marked out of 7 and rounded to the nearest half mark.

While the exact matches are as good as with the first corpus, having 80.4% of cases differing by no more than 0.5 is not as good but is still reasonable. The number of poor results where the difference was 1.5 or more was low at 2%.

Table 4 Differences between the automatic marker and 'true' marks (second corpus)

Difference	**0**	**0.5**	**1**	**1.5**	**2**	**>2.0**
Number	45	33	13	4	2	0
%	46.39	34.02	13.4	4.12	2.06	0
Cum. %	48.91	80.41	93.81	97.93	100	100

AC1 and kappa values were calculated for this corpus, in the same way as for the first corpus. These results are shown in Table 5. As can be seen, the agreement between all pairs of markers is poor. In this corpus, many of the marks awarded by different markers differed by 0.5 marks. To accommodate these small (generally considered trivial) differences, we rounded all the marks to the nearest whole mark and recalculated the agreement measures. These are shown in Table 6. As with the first corpus, the agreement measures show that the markers agree, to some extent, on the marks that should be awarded to each diagram and we can reject the null hypothesis that some of the marks are awarded randomly. Again, we can see that on the testing set the automatic marker agrees with the 'true' marks more strongly than do the first marks. Therefore, we again conclude that the automatic marker is a more accurate marker than an arbitrary human marker. Notably, the automatic marker performs better on the unseen testing set than it does on the testing set!

Table 5: Inter-marker agreement for the second corpus

	True - First		**True - Auto**		**First - Auto**	
	AC1	**Kappa**	**AC1**	**Kappa**	**AC1**	**Kappa**
Training set, half marks	0.2756	0.2230	0.2005	0.1481	0.0379	-0.0140
Testing set, half marks	0.2289	0.1875	0.4023	0.3651	0.2182	0.1741

Table 6: Inter-marker agreement for the second corpus; marks rounded to the nearest whole mark.

	True - First		**True - Auto**		**First - Auto**	
	AC1	**Kappa**	**AC1**	**Kappa**	**AC1**	**Kappa**
Training set, whole marks	0.5358	0.4658	0.6291	0.5684	0.3674	0.2845
Testing set, whole marks	0.3924	0.3106	0.5984	0.5270	0.3811	0.2917

A comparison of the results across the range of marks for the second corpus experiment is shown in Figure 8.

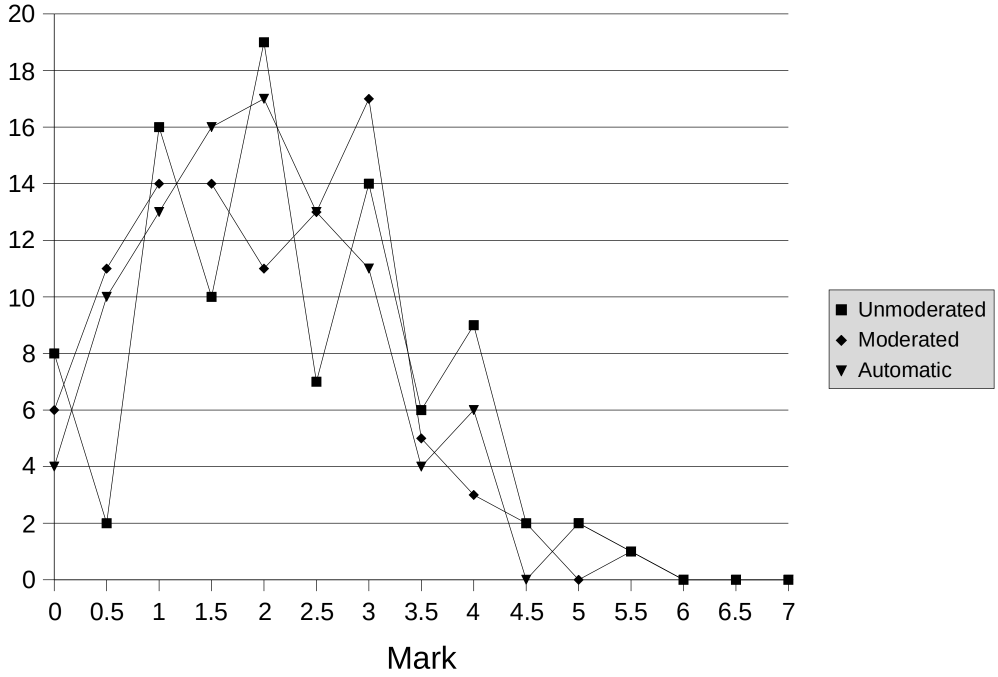


Figure 8 Comparison of marks across the range (second corpus)

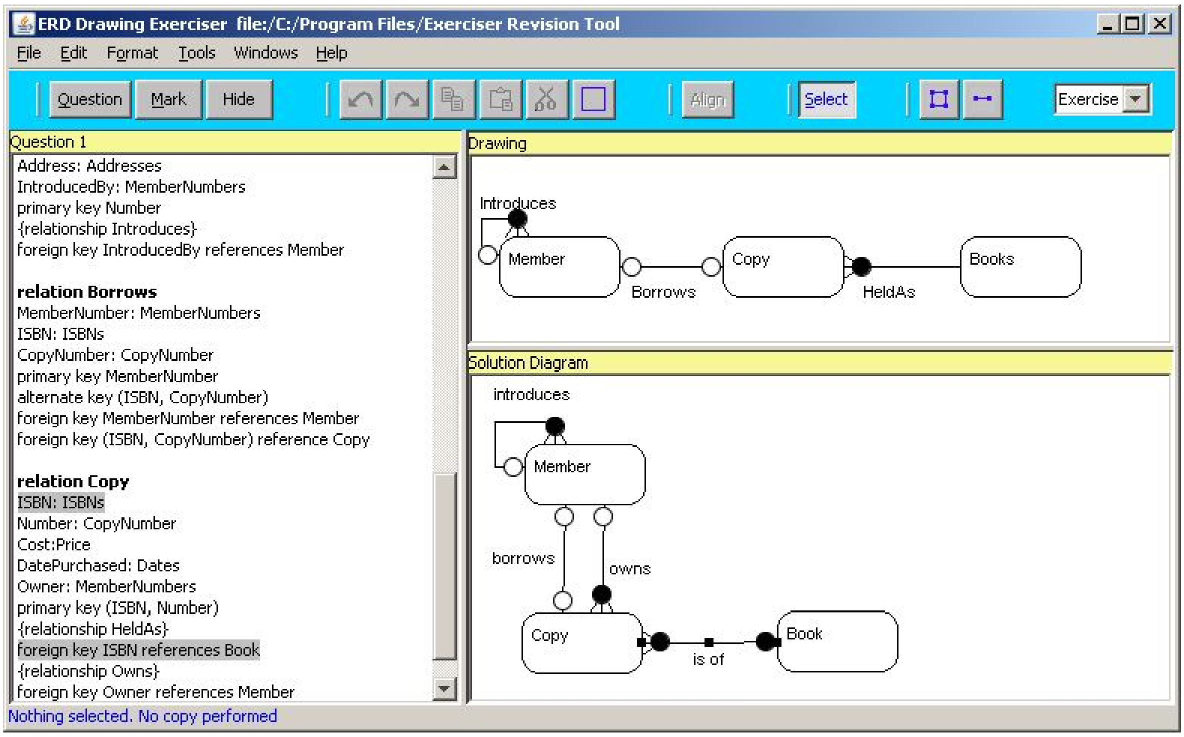
As with the first corpus, there is reasonable agreement across the range of marks. Nevertheless, it is quite clear that the students' answers tended to be quite poor with no-one gaining full marks. The range of marks awarded both by the humans and the automatic marker is much smaller than in the previous experiments. The mean difference between the 'true' human mark and the automatic marker's marks was 0.397 with a standard deviation of 0.486. In those cases where the automatic marker did not agree with the 'true' human marks, the automatic marker was more generous than the human marker in 21 cases, and less generous in 26 cases again showing little bias.

Armed with this information, we felt confident that the technology could be incorporated into a number of software tools to support learning and assessment, particularly in a formative environment.

5. Tools for learning: Revision tool

5.1. Revision tool

The revision tool, details of which can be found in (Thomas, Waugh and Smith 2007a), whose user interface is shown in Figure 9, is aimed at providing students with an opportunity to practice drawing ERDs in response to typical assessment and examination questions. The tool is delivered pre-loaded with several questions (we currently provide 10) in textual form (see the left-hand pane in Figure 9). The student draws their diagram in the top-right-hand pane. Once satisfied with their diagram, the student can press the 'Mark' button to request the tool to mark the diagram and provide feedback.



Address: Addresses
IntroducedBy: MemberNumbers
primary key Number
{relationship Introduces}
foreign key IntroducedBy references Member

relation Borrows
MemberNumber: MemberNumbers
ISBN: ISBNs
CopyNumber: CopyNumber
primary key MemberNumber
alternate key (ISBN, CopyNumber)
foreign key MemberNumber references Member
foreign key (ISBN, CopyNumber) reference Copy

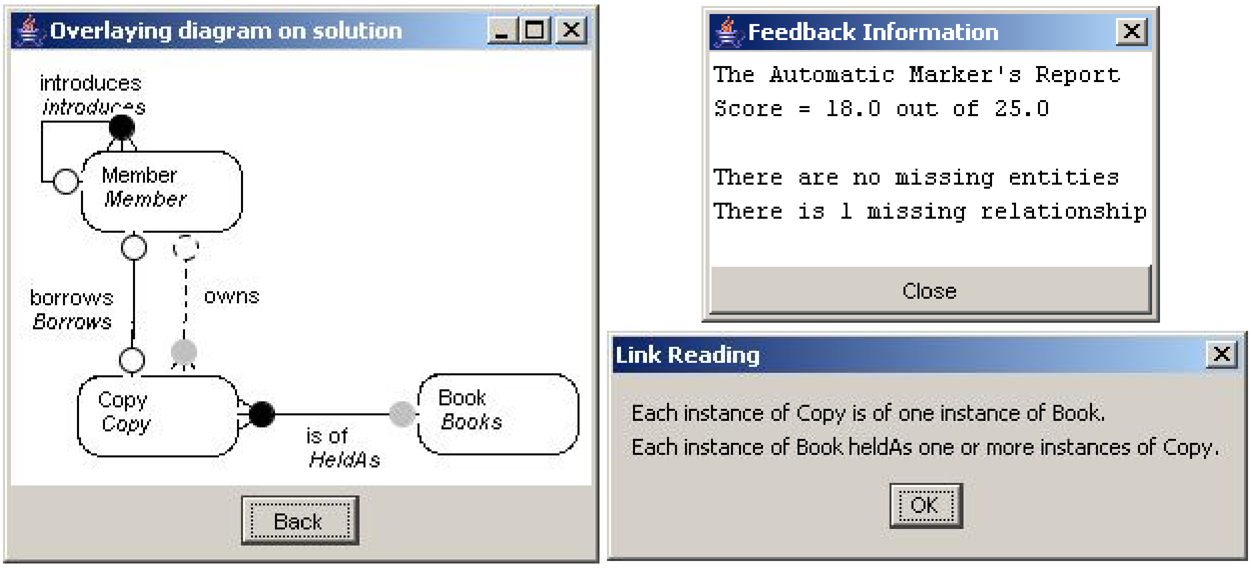
relation Copy
ISBN: ISBNs
Number: CopyNumber
Cost:Price
DatePurchased: Dates
Owner: MemberNumbers
primary key (ISBN, Number)
{relationship HeldAs}
foreign key ISBN references Book
{relationship Owns}
foreign key Owner references Member

Nothing selected. No copy performed

Figure 9 The revision tool

Feedback on the student attempt is provided in a number of ways. The diagram in the bottom right-hand pane, blanked-out when the student starts a new question, is the model answer which is revealed by pressing the 'Show/Hide' button. All the features of the model answer diagram are selectable. When the user selects a feature (the relation 'is of' is shown selected in Figure 9), those parts of the question text that relate to the feature are highlighted. The user can also right-click on a feature to obtain an explanation of how that feature relates to the problem.

When the student attempt is marked, the first element of feedback is the automatically generated mark and an identification of missing elements. The student can then request a comparison of their attempt with the model answer shown graphically (see Figure 10). It is also possible to obtain a description or 'reading' of each link in the model answer. More details can be found in (Thomas, Waugh and Smith 2007a).



The Automatic Marker's Report
Score = 18.0 out of 25.0

There are no missing entities
There is 1 missing relationship

Each instance of Copy is of one instance of Book.
Each instance of Book heldAs one or more instances of Copy.

Figure 10 Various items of feedback from the revision tool

The revision tool has been tested with two sets of student volunteers (a total of 41 students) and found to be a very welcome addition to our database course. All testers found the tool either easy or fairly easy to use (measured on a four-point Lickert scale). All but one student said that they would recommend the tool to other students.

The revision tool requires a considerable amount of data to be generated by the instructor, including the questions, model answers and marking schemes. We have developed an authoring tool (Waugh, Thomas and Smith 2007a) that assists with these tasks.

5.2. Assessment tools

The automatic marker is available as a stand-alone tool primarily designed for batch processing of large numbers of diagrams and providing statistics, though it is possible to mark diagrams individually. In its batch processing mode, it is capable of providing the teacher with useful feedback on the performance of students as a group. For example, it can show the frequency with which MMUs were found to be missing from student answers.

In practice during marking, several alternative solutions (not all of which need to be entirely accurate, but which deserve some credit) often come to light. Depending at what stage the marking has reached and the size of the cohort, such discoveries can be a significant overhead in the marking process since answers that have already been marked have to be reviewed. With an automatic marker it is a straightforward and speedy matter to draw the alternative solution and associate a mark scheme with it (using the authoring tool, for example), and then let the automatic marker compare each answer against all solutions and choose the appropriate mark. Our automatic marker assumes that, in general, there will be multiple solutions to a question.

We envisage that the automatic marker could be used in a number of ways:

1. as a 'second marker' in which its results are compared to those of a human marker. In cases where there is a significant discrepancy between the human and the automatic marker, a second human opinion can be requested and hence reduce the overhead in 'double marking'

by humans.
2. the results of the automatic marker can be used to identify those students close to a border line which can be re-marked by a human;
3. as a guide or check on the grading performed by a single human marker as is normally the case with course work.

An examination of those cases where there was a significant discrepancy between the human and automated marking, confirming the findings of (Hungerford, Hevner and Collins 2004) that even experts in the field can misinterpret diagrams. In many of these cases it was difficult to decipher the student diagram because its structure was quite different to that of the model answer. We felt that the layout of the student answer was making it difficult for the human marker to recognize where marks should be awarded. Therefore, we have developed a ‘marker’s assistant’ that redraws a student’s answer diagram in a form that more closely resembles the model answer making the diagram easier to assess. Figure 11 shows two screen shots of the interface to the prototype assistant.

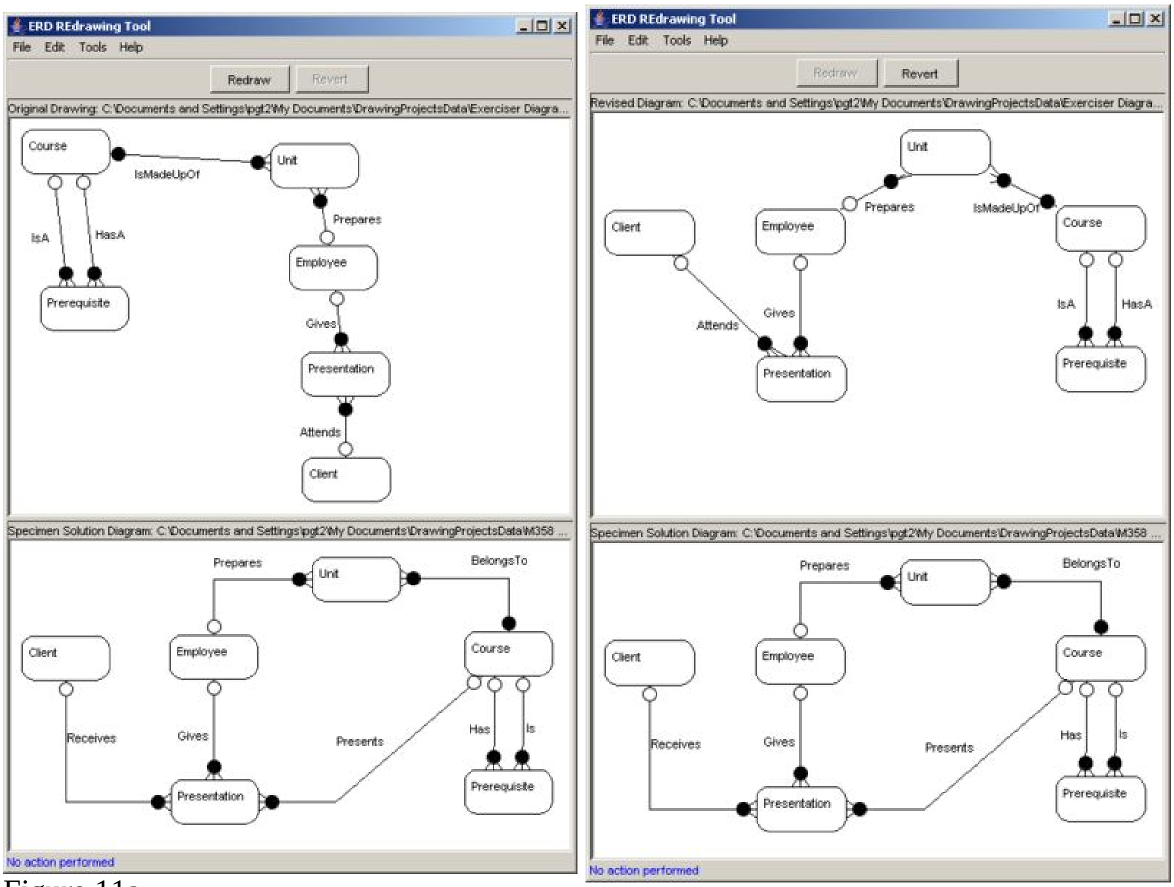


Figure 11a

Figure 11b

Figure 11 The marker’s redrawing assistant

The redrawing feature can also be used ‘in reverse’. That is, it can redraw the model answer in a form that more closely resembles the student’s diagram. We think that in this mode, the tool could be an additional source of feedback to students. Students will be more easily able to comprehend the model answer when it is presented in a form that corresponds to their own diagram.

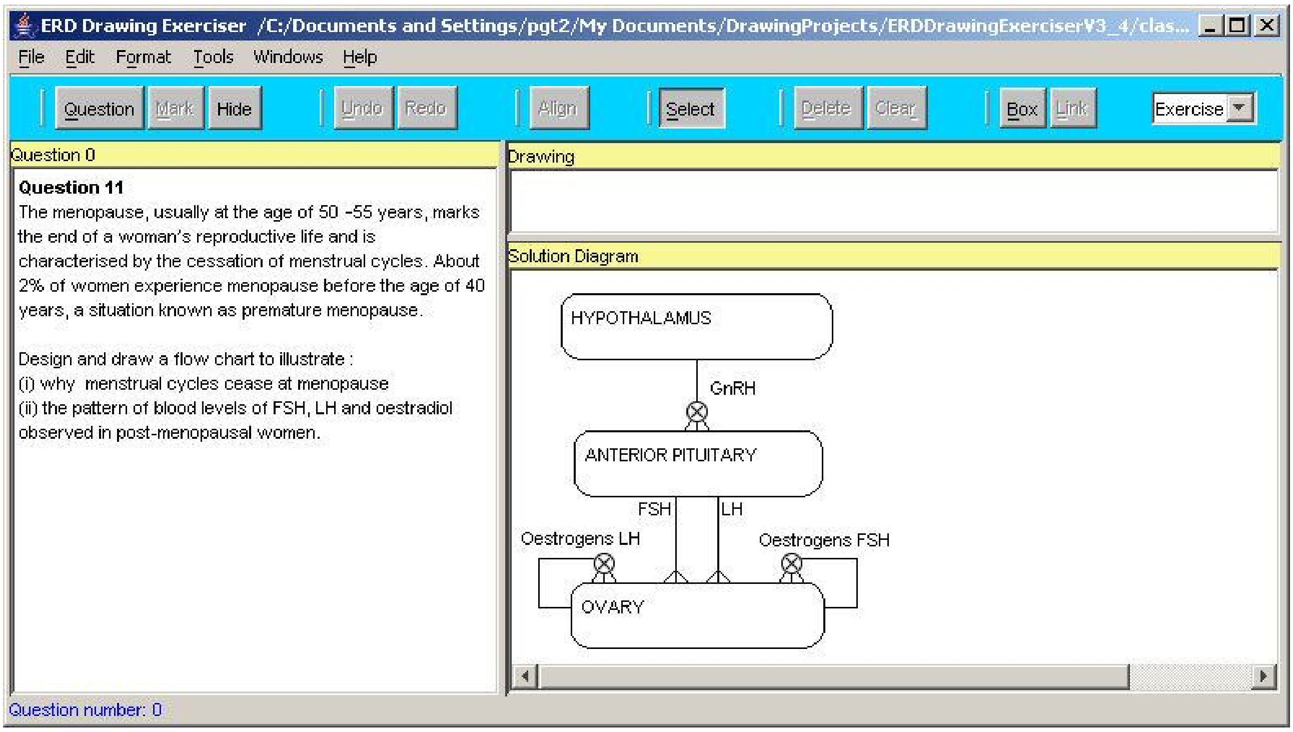
6. Conclusion and future work

The experiments we have performed with the automatic marking tool have been encouraging. We have a framework for capturing, processing and interpreting graph-based diagrams that has worked well in a practical situation. The interpretation phase has been applied to the marking of ERDs and compared with human markers has been shown to be effective across a large number of ER diagrams. Not only can the algorithm be used for grading diagrams, it has also been incorporated into three teaching and learning tools. One of the tools, a tool for revising ERDs, has been tested with students and found to provide useful support on a database course.

There are a few diagrams in our corpora that are not dealt with sufficiently well by the automatic marking algorithm. In some cases, the deficiency can be attributed to the lack of a more global view of the diagrams. Currently, the marking algorithm only searches 'locally' for information: either the immediate context of an entity or adjacent relationships that might be the result of a many-to-many relationship expansion. That is, the current algorithm focuses mainly on MMUs not the larger MUs that might be present in the diagrams. Incorporating knowledge about aggregate equivalences and the way in which one instance of an MMU could contribute useful information when attempting to match other instances of an MMU should improve the performance. A start has been made on this aspect and is reported in (Thomas, Waugh and Smith 2006).

We would also like to improve some aspects of its performance by increasing the use of further natural language processing (NLP) algorithms (Manning and Schutze, 2002).

We have begun to investigate the use of the technology in other domains. The revision tool has been modified in a minor way to allow it to be configured with different diagramming notations while retaining its fundamental graph-based properties. Figure 12 shows its use for revising biological flows. In this domain, the weights must be set to reflect the relative importance of the flows to the biological entities.



Question 11
The menopause, usually at the age of 50 -55 years, marks the end of a woman's reproductive life and is characterised by the cessation of menstrual cycles. About 2% of women experience menopause before the age of 40 years, a situation known as premature menopause.

Design and draw a flow chart to illustrate :
(i) why menstrual cycles cease at menopause
(ii) the pattern of blood levels of FSH, LH and oestradiol observed in post-menopausal women.

Figure 12 The use of the revision tool in the biological flow domain

We have also embarked on building a corpus of UML sequence diagrams with a view to widening the applicability of our approach since such diagrams not only possess MMUs similar to

ERDs (objects related to one another through message sending) but also require the notion of sequence (of messages/relationships) to be dealt with effectively.

Finally, there is work to be performed in relation to the robustness of the automatic marker (identifying the extent to which a user could 'fool' the tool into giving high marks for poor answers) and the steps that might be taken to prevent misuse, particularly if it were to proposed to use the tool in a high-stakes environment such as an examination.

7. Acknowledgement

Part of this work was undertaken while one of the authors held a HEFCE funded Teaching Fellowship of COLMSCT (Centre for Open Learning in Mathematics, Science, Computing and Technology) Centre of Excellence for Teaching and Learning (CETL) at the Open University.

8. References

Anderson, M., B. Meyer, P. Olivier (eds) 2002. *Diagrammatic Representation and Reasoning*, London: Springer-Verlag.

Anderson, M., R. McCartney. 2003. Diagram processing: Computing with Diagrams. *Artificial Intelligence* **145** (1-2): 181—226.

Batmaz, F. and C.J. Hinde. 2006. A Diagram Drawing Tool for Semi-automatic assessment of Conceptual Database Diagrams. Proceedings of the 10th Annual International Conference in Computer Assisted Assessment. (Loughborough University, UK, July 2006), 68—81.

Batmaz, F. and C.J. Hinde. 2007. A Web-Based Semi-Automatic Assessment Tool for Conceptual Database Diagrams. Proceedings of the Sixth IASTED International Conference on Web-Based Education, Chamonix, France, March 14-17, 2007, 427—432.

Burstein, J., M. Chodorow and C. Leacock. 2003. Criterion SM Online Essay Evaluation: An Application for Automated Evaluation of Student Essays. Proceedings of the Fifteenth Annual Conference on Innovative Applications of Artificial Intelligence, (Acapulco, Mexico. August 2003.

Chok, S.S. and K. Marriott. 1995. Parsing visual languages. in Proceedings of the Eighteenth Australian Computer Science Conference, *Australian Computer Science Communications* **17**: 90—98.

Fleiss, J. L. 1971 Measuring nominal scale agreement among many raters. *Psychological Bulletin* **76**(5): 378—382

Flower, J. and J. Masthoff and G. Stapleton. 2004. Generating Readable Proofs: A Heuristic Approach to Theorem Proving with Spider Diagrams, in Blackwell, A.; Marriott, K. & Shimojima, A., (eds.), *Diagrammatic Representation and Inference*, LNCS Vol. 2980, Springer., pp. 166—181.

Gwet, K. (2001) *Handbook of Inter-Rater Reliability*, Gaithersburg: StatAxis Publishing

Haley, Debra Trusso, Pete Thomas, Anne De Roeck, and Marian Petre. (2005). A Research Taxonomy for Latent Semantic Analysis-Based Educational Applications. In G. Angelova, K. Bontcheva, R. Mitkov, N. Nicolov & N. Nikolov (eds.), International Conference on Recent Advances in Natural Language Processing'05. (Borovets, Bulgaria, 2005), 575—579.

Hammond, T. and Davis, R. (2005) LADDER, a sketching language for user interface developers. *Computers & Graphics*, **29**, 518-532.

Higgins, C. A. and B. Bligh. 2006. Formative Computer Based Assessment in Diagram Based Domains. Proceedings of the 11th Annual Conference on Innovation and Technology in Computer Science Education (ITiCSE, Bologna, Italy, June 26-28, 2006), 98—102.

Hungerford B. C., A. R. Hevner and R.W. Collins. 2004. Reviewing Software Diagrams: A Cognitive Study, *IEEE Transactions on Software Engineering* **30** (2) Feb 2004 82—96.

Kniveton, B. H. A. 1996. A correlational analysis of multiple-choice and essay assessment measures. *Research in Education*, **56**, 73—84.

Manning, C.D and H. Schutze. 2002. *Foundations of Statistical Natural language Processing*. MIT Press, Cambridge, Massachusetts, USA.

Marriott, K. and B. Meyer. 1998. *Visual Language Theory*. Springer-Verlag, New York

Marriott, K., B. Meyer and K.B. Wittenburg. 1998 A survey of Visual Language Specification and Recognition. In *Visual Language Theory*, eds: K. Marriott and B. Meyer. Springer-Verlag, New York, 8-85.

Newstead, S.E. and I. Dennis. 1994. The reliability of exam marking in psychology: examiners examined. *Psychologist*, **7** (5) 216—219.

Shermis, M.D, J.C. Burstein. 2003. (eds.) *Automated Essay Scoring: a cross-disciplinary approach*. Lawrence Erlbaum Associates, Mahwah, NJ, USA.

Smith, N, P.G. Thomas and K. Waugh. 2004. Interpreting Imprecise Diagrams. Proceedings of the Third International Conference in the Theory and Application of Diagrams. March 22-24, Cambridge, UK. Springer Lecture Notes in Computer Science, eds: Alan Blackwell, Kim Marriott, Atsushi Shimojima, 2980, 239—241.

Thomas, P.G. 2003. Evaluation of Electronic Marking of Examinations, Proceedings of the 8th Annual Conference on Innovation and Technology in Computer Science Education (ITiCSE, Thesaloniki, Greece, 2003), 50—54

Thomas, P.G., K. Waugh and N. Smith. 2005. Experiments in the Automatic marking of E-R Diagrams. Proceedings of the 10th Annual Conference on Innovation and Technology in Computer Science Education (ITiCSE, Monte de Caparica, Portugal, 2005), 158—162.

Thomas, P.G., K. Waugh and N. Smith. 2006. Using Patterns in the Automatic Marking of ER-Diagrams. Proceedings of the 11th Annual Conference on Innovation and Technology in Computer Science Education (ITiCSE, June 26-28, Bologna, Italy, 2006), 403—413.

Thomas, P.G., K. Waugh and N. Smith. 2007a. Tools for supporting the teaching and learning of data modelling. Proceedings of ED-MEDIA conference (ED-MEDIA, 25—29 June 2007, Vancouver, 2007) 4014—4018

Thomas, P.G., K. Waugh and N. Smith. 2007b. Computer Assisted Assessment of Diagrams. Proceedings of the 12th Annual Conference on Innovation and Technology in Computer Science Education (ITiCSE, June 25-29, Dundee, Scotland, 2007), 25—29.

Thomas, P.G., K. Waugh and N. Smith. 2007c. Learning and automatically assessing graph-based diagrams. In S. Wheeler and N. Whitton (eds). Beyond Control: learning technology for the social network generation. Research Proceedings of the 14th Association for Learning Technology Conference (ALT-C, 4-6 September, Nottingham, UK, 2007), 61-74.

Tsintsifas A. (2002) *A Framework for the Computer Based Assessment of Diagram-Based Coursework*, Ph.D. Thesis, Computer Science Department, University of Nottingham, UK.

Waugh, K.G., P.G. Thomas and N. Smith. 2004. Toward the Automated Assessment of Entity-Relationship Diagrams. Proceedings of the 2nd LTSN-ICS Teaching, Learning and Assessment in Databases Workshop, (TLAD, July 2004, Edinburgh).

Waugh, K.G, P.G. Thomas and N. Smith. 2007. Teaching and learning applications related to the automated interpretation of ERDs. Proceedings of the 5th LTSN-ICS Teaching, Learning and Assessment in Databases Workshop (TLAD , July 2007, Glasgow, UK).