DOMAIN-SPECIFIC FORMATIVE FEEDBACK THROUGH DOMAIN-INDEPENDENT DIAGRAM MATCHING

Christos Tselonis
John Sargeant

Domain-Specific Formative Feedback through Domain-Independent Diagram Matching

Christos Tselonis, John Sargeant
School of Computer Science
University of Manchester
{tselonic,johns}@cs.man.ac.uk

Abstract

As part of our Human-Computer Collaborative (HCC) approach to assessment, we seek representations of answers and marking judgements which can be applied to a wide variety of situations. In this paper we introduce such a representation, which we call a *gree*[1], and discuss an initial practical application of grees for formative feedback. An experiment was carried out in which students were asked to construct an answer while receiving interactive feedback and then complete a short survey. The results show that it is possible to give effective domain-specific formative feedback based on a domain-independent internal representation or "metaformat".

This work builds on results we have previously presented on domain-independent diagram matching based on heuristic matching of graphs. Grees provide much greater flexibility, with a wide variety of potential applications. We discuss some problems which need to be overcome before we can realise their full potential.

Introduction

Fully automated marking for constructed answers such as diagrams and text is a very difficult task. Although there are implementations attempting to generalise the marking process [2], most efforts focus on single knowledge domains or depend on particular semantics [1, 8, 9, 11], lacking reusability and extendibility.

We have proposed the human-computer collaborative (HCC) approach as a solution [7], according to which marking is a dynamic process where the computer deals with repetitive tasks while the human makes the important judgements. We have shown that such an approach can significantly reduce the effort and the time taken for a human to mark a large number of answers.

[1] As part of the commercialisation of the ABC software by Assessment21 Ltd., the use of grees in assessment is the subject of a patent application.

In parallel, we now attempt to enhance the student experience, by extending the system to dynamically generate real time feedback, based on matching the student's answer against a model answer.

The story so far

In [10] we discussed a way of matching constructed answers, and in particular diagrams, based on heuristics. The method involves the conversion of answers into enriched graphs, whose components (which we call "boxes", and "connectors") retain some of the attributes existing in the original diagrams, such as types and labels (text strings associated with the boxes and connectors). Feeding a model graph and any number of student graphs into the matching mechanism, along with a set of metric / weighting pairs determining the matching process, results in a number of local scores associated with the graphs' nodes, which eventually are combined to produce a score of similarity between the graphs.

The similarity scores not only resemble -in most cases[2]- the marks previously awarded by a human marker, but most importantly could provide the means to improve consistency and minimise marking time; sorting the answers by similarity to the model answer or viewing these similarities highlighted in colour certainly helps in this respect.

In the next section we explain grees and gree matching. Then we describe an encouraging experiment in using grees for formative feedback. Finally we draw conclusions and discuss a number of further issues.

Grees and the matching mechanism

The revised matching mechanism, although based on the one introduced in [10][3] includes significant enhancements; it is now extended to adopt a *modular scoring strategy*, according to which *parts* of the model answer are separately matched to parts of the candidate answer. This way, marking schemes can be accurately defined and marks awarded for the parts of the answer that really matter, although they can be dynamically amended later on if necessary. Different parts of the model answer may be worth a different portion of the total marks available, and can also be weighted differently, according to their components' relative importance. Equally importantly, multiple alternative acceptable parts deserving the same portion of marks can be set for a single constructed answer.

To enable this modular approach, *grees*, **dynamically extendable AND/OR trees whose leaf nodes are overlapping graph fragments**, were invented. They effectively represent the model answer parts along with any marking

[2] Cases where student drawings abided by some basic rules, e.g. an answer should be a single connected graph, box labels should be placed *in* the box, not above it etc.

[3] A number of details, explained in [10] are omitted from the description here, so readers desiring a complete account of the matching process will need to consult that paper

judgements and other information needed in a systematic manner, allowing for reusability, extendibility and modularity. Although some aspects of grees, such as the use of AND/OR trees to represent marking schemes, have been proposed before, the combination is, to the best of our knowledge, novel.

Being a generic metaformat, grees do not depend in any way on the knowledge domain of the question; as long as an answer can be converted to a graph consisting of boxes and connectors, any answer type may be modelled by a gree, including diagrams, mathematical expressions, software programs and even short, factual text fragments. No domain-specific information is contained in a gree, or used by the matching mechanism.

The matching process takes place between a model answer stored in the gree metaformat and a set of candidate answers converted to graphs as shown in Figure 1.

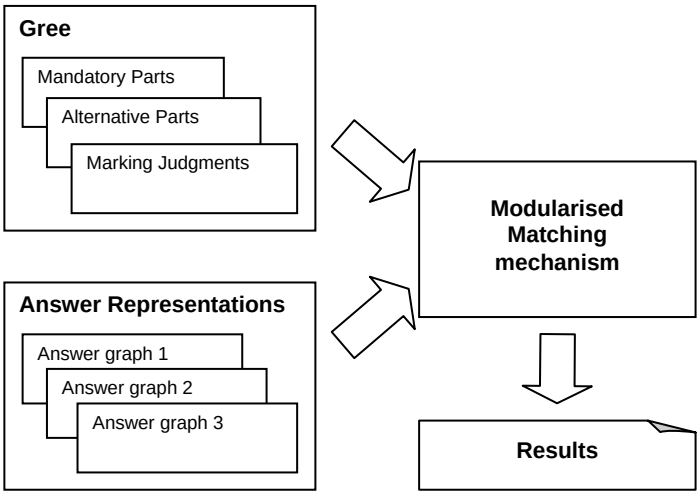

Figure 1: the matching process.

Grees in detail

Our example is based on the question in listing 1, set by the second author for a software engineering examination in January 2006, which requires students to draw a UML class diagram. Although not trivial, the question tightly constrains what a correct answer must look like. Figure 2 shows two possible fully correct answers. They include several different components (circled), but also some spatial differences, which are ignored by the matching mechanism.

You are designing an online book information system for the Resource Centre. This will allow students to find out about which books, or chapters of books are recommended for each course, and also to read or write reviews of books or chapters for the benefit of other students. The software will also attempt to provide a summary of each Chapter. You have identified the domain classes Book, Chapter, Course, and Review, and there will be corresponding design classes. Since Chapters as well as complete Books may be recommended or reviewed, you have added an additional design class ReadingMaterial to capture the common properties of the two. Draw a skeleton design class diagram to show the exact relationships between these five classes (but not their attributes or operations).

Listing 1: The examination question.

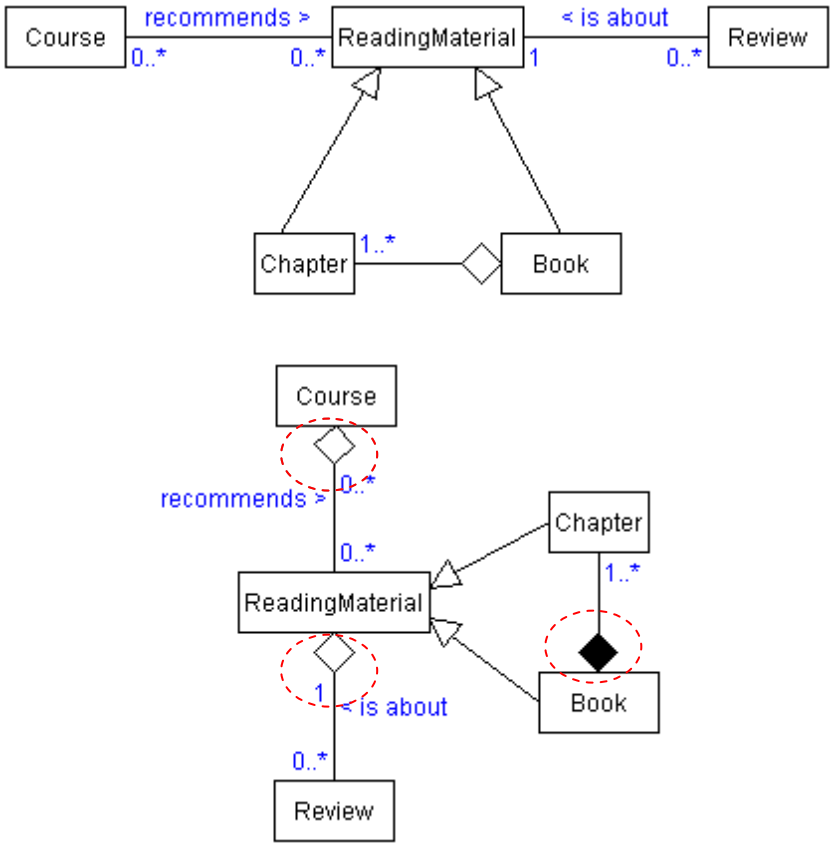

Figure 2: Two of the possible fully correct answers.

Figure 3 represents a gree which describes 8 fully correct answers to the question. Its leaf nodes, labelled A to H, contain possible parts of the correct answer. Each part answer comprises a portion of an acceptable class diagram, in combination with a number of parameters describing how the answer part should be matched. In particular, the parameters include the number of marks allocated for the answer part, the generic metrics considered during the matching process for the various components, and features comprising the answer part, weightings specifying by how much the metrics should count towards the final score and flags determining whether these

metrics should contribute to both the matching and the final score, or just the former, in order that the algorithm matches corresponding parts correctly.

For example, consider node A in Figure 3. In terms of the question, this specifies one of two possible correct ways of representing the relationship between the Chapter and Book classes (node B being the other). In particular the black diamond, representing a strong association, is important.[4]

The algorithm must first ensure it is matching the correct part of the graph, in this case the boxes labelled Chapter and Book and the connection between them. So for instance the box label metric has the maximum weight of 4. Having made the correct match, for scoring purposes we only care about the type of connector and its label (as indicated by the ticks in the ”check boxes”). Note that between the previous paragraph and this one we have moved from domain-dependent concepts to a domain-independent algorithm.

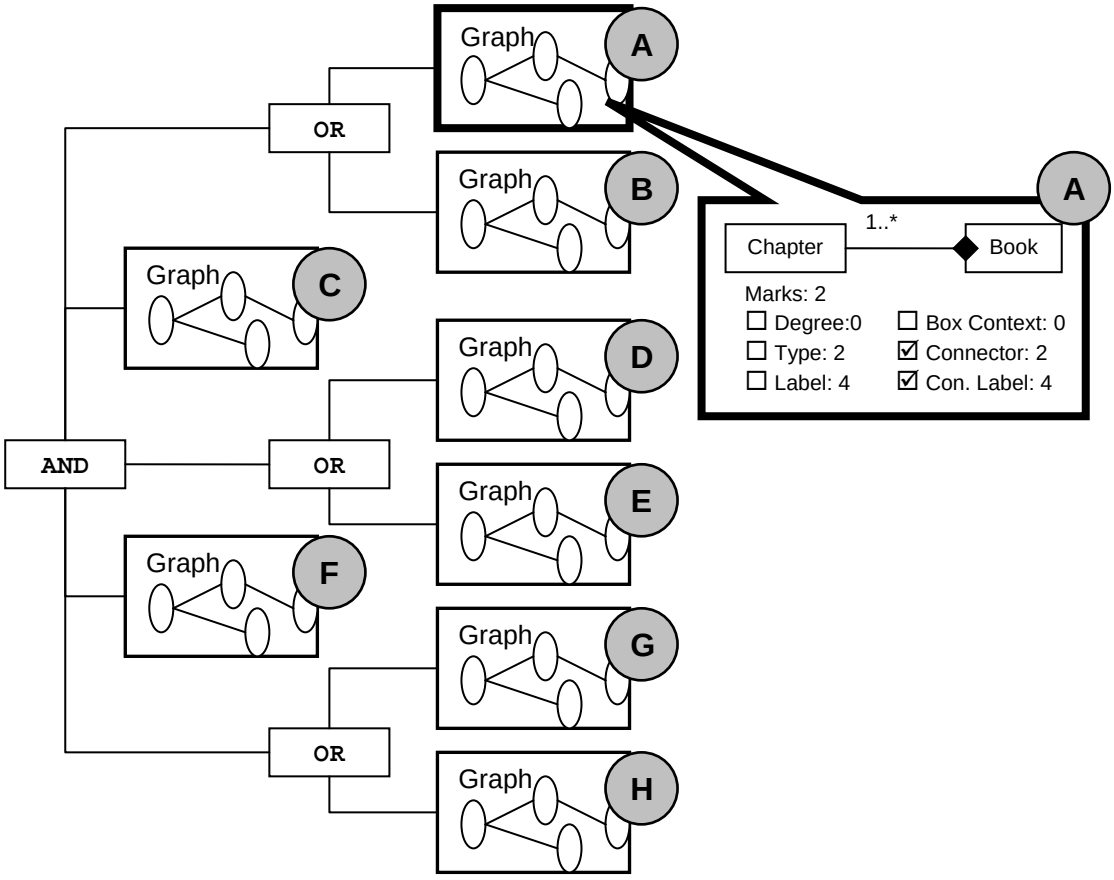

Figure 3: A gree representing a set of correct answers.

[4] Although the first author’s original marking scheme only allowed for the option represented by node B, an example of the HCC principle that marking schemes usually need to be extended dynamically on the basis of student answers.

The leaf nodes are connected in a tree structure by AND and OR nodes. For a submitted answer to be awarded full marks, it must contain all the sub-parts specified by the subtree below an AND node. For a group of leaf nodes placed directly under an OR node, the content of only one of them need form part of the submitted answer. Clearly in this example, a submitted answer worth full marks must contain (A OR B) AND C AND (D OR E) AND F AND (G OR H). This gree fully describes $2^3 = 8$ alternative and fully correct answers.

In order to match a submitted answer, such as either of those shown in Figure 2, against a gree, the matching mechanism starts by considering the gree's root node and continues traversing the nodes down the tree. Once a leaf node is encountered, i.e. a node that contains a graph fragment, a score for that node compared to the submitted answer is calculated using heuristic methods as explained in [10]. For all nodes descending from an OR node, the one producing the highest score is considered to be the closest match. This score is thereafter the one associated with that OR node. For all nodes depending directly from an AND node, the scores are added. Once the matching process has completed, the score given by the root node is the mark awarded to the submitted answer.

Theoretically, grees can be re-adjusted on the fly during the marking process in the light of previously unconsidered alternative correct parts of submitted answers. This could involve adding more nodes, reconnecting existing ones differently, splitting the marks differently, or changing values for the metrics. The system will then automatically recalculate the scores for the already marked answers and notify the human marker for the submissions whose scores have changed. The gain can be significant over traditional paper based marking, where changes to the marking scheme part way through a large number of submissions require reviewing all answers marked so far. Although the gree specification supports it, a tool which would allow end-users (i.e. markers) to edit grees has not yet been developed, since the user interface issues are significant. However, an experimental editor application exists (Figure 4), which may form the basis of a marking tool in the future.

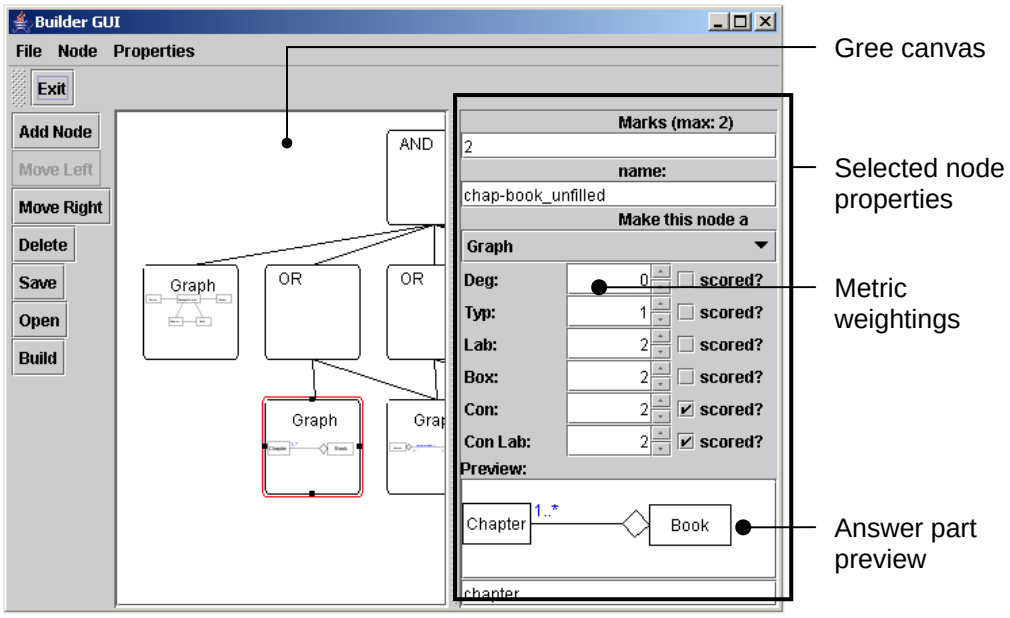

Figure 4: A basic gree editor.

According to the HCCA paradigm, the human marker is responsible for affirming or amending the automated marking results. To ease this process, a marking tool based on grees could support the visual features discussed previously [10]; a part of the submitted answer can be highlighted with the same colour as a gree's node, to indicate a match (Figure 5A). Alternatively, the contents of the gree's nodes, which are live graph objects, may be coloured according to matched parts of the submitted answer (Figure 5B).

Additionally, sorting the submitted answers by mark, status (marked / unmarked), completeness (number of gree nodes matched) etc is a straightforward extension.

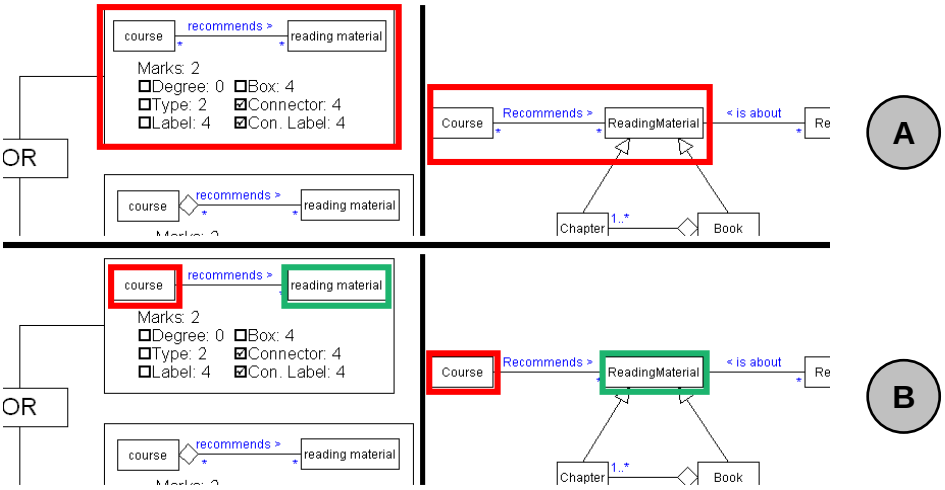

Figure 5: Communicating the matches visually.

Another important feature of the gree is the ability to reconstruct programmatically the full set of possible correct answers. In addition they provide the ability to determine which of these correct answers is the closest to another answer. This is important when using grees to provide students with instant feedback, for example during formative or self assessment

Use of grees in formative assessment

As a first practical application, a standalone tool intended for formative and self assessment [2, 3, 5, 6], taking advantage of the gree matching mechanism, was developed[5]. It displays a question to a student, allowing them to draw the answer, while providing automated feedback. The tool translates the results of matching the student's current drawing against a gree into meaningful feedback strings. The strings, which may vary from general hints to very specific information such as suggested content and component locations, are displayed on the drawing canvas via popups. Listing 2 displays a number of example feedback strings.

- A better label for this box might be *'Course'*.
- This box's type seems incorrect (Should be '*Class'*).
- There are *2* boxes too many connected to this one.
- There should be *2* more boxes connected to this one.
- This connector's type seems incorrect.
- This connector's direction seems incorrect.
- This connector should have at least one more label (possible valid position marked with 'x').
- This connector has at least one label too many.
- One or more of this connector's labels are misplaced (possible valid position marked with 'x').

Listing 2: Example feedback strings.

Figure 6 displays the feedback tool in action. Hovering over a popup "activates" it, highlighting the popup as well as the component it refers to. Clicking on it causes it to be dismissed. An extra button to clear all popups at once is also available.

[5] Based on a diagram drawing tool initially developed by Stuart Anderson.

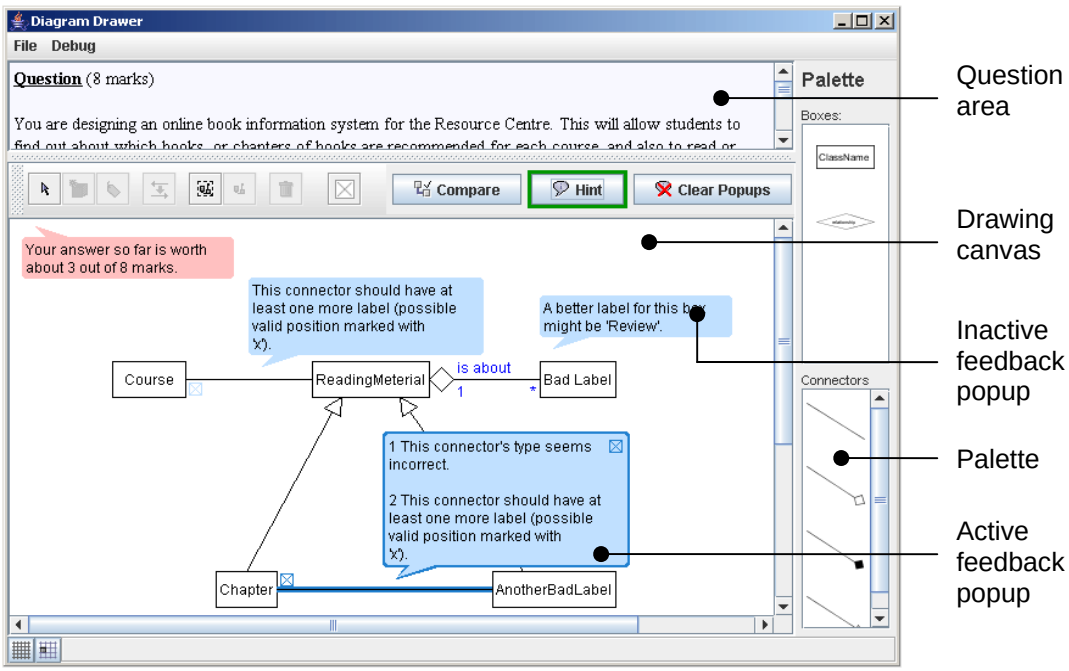

Figure 6: The feedback tool.

The tool also incorporates a 'Compare' button which when pressed, will query the system for the fully correct answer which is closest to the current drawing. It will then display both side by side in a new window. Figure 7A displays the closest correct dynamically reconstructed answer to a student answer shown at the top, while Figure 7B displays the same diagrams, with two of the student answer's boxes omitted. In this case, the closest answer looks somewhat unbalanced. Students, however, were able to drag the boxes around to make the diagram clearer.

The experiment

Second year Computer Science students attending the '*COMP2341: Software Engineering I*' module were asked to take part in this experiment, evaluating gree matching and feedback generation. The feedback tool was deployed as a Java applet, capable of running over the Internet on any Java-enabled browser, so students could run it in their own time, completely anonymously. They were also given the option of a supervised session following an exam revision class, but none made use of this opportunity, possibly because it was four whole days before the exam. Therefore all students who participated did so with no help or supervision.

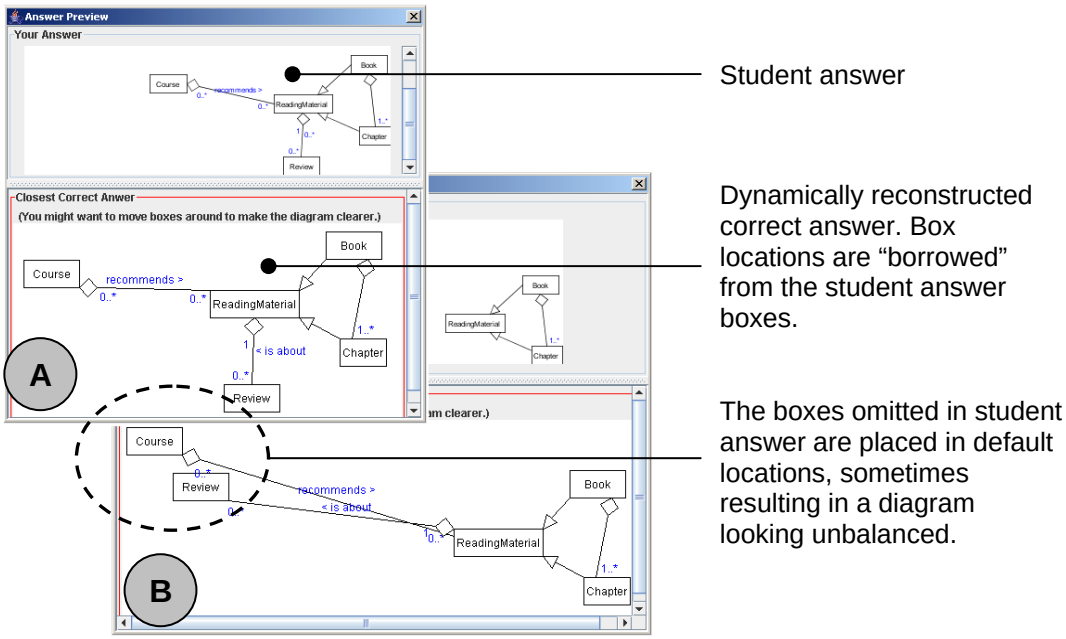

Figure 7: Dynamically reconstructed correct answers.

First, the trial featured a short tutorial session during which students had the opportunity to familiarise themselves with the tool and the feedback mechanism, by following a series of step-by-step instructions in order to construct a trivial diagram. During the tutorial, the students were guided to intentionally make errors so the feedback features, triggered automatically upon their actions, were emphasised.

They were then presented with the main question, (Listing 1) from the previous year's examination, asking them to draw a UML class diagram like those shown in Figure 2. Both the feedback and the 'Compare' buttons could be used at any time, any number of times. However, all such interaction was being recorded and when viewing the closest fully correct answer for comparison, editing the answer was disabled.

Once a student elected to commit to their final answer, they were presented with a short, optional, survey, assessing the tool's usefulness. The survey responses along with the diagram answer and the statistical data were finally submitted back to the server.

The results

A total of 42 submissions were received, two of which contained no usable data. Although there was no definitive way to determine whether all submissions were submitted by different users because of complete anonymity, it is likely that all or most were, judging from the differences between the answers and the submission timestamps.

The first question of the survey was "*How many times did you use the hint mechanism?*". The students could enter an integer in a spin edit control, or leave the default 0. The number of times the feedback mechanism was *actually* used was recorded and ranged from 0 to 60 per submission. The difference between the survey responses (estimates) and the actual number of times was great, both overall and on a per student case; generally, students tended to underestimate this number by about a factor of 2. Table 1 summarises the estimated and actual ranges. For instance 25 students believed they had used the feedback mechanism no more than 4 times, but only 9 had actually done so.

Times Used	Submissions	
	Actual	Estimate
0 - 4	9	25
5 - 9	12	6
10 - 14	3	3
15 - 19	2	4
20 - 24	5	0
25 - 29	3	0
30+	6	2

Table 1: Actual and estimated number of times the feedback mechanism was used per submission.

The second survey question was "*How clearly was the feedback information presented?*". The students could select one of four options, shown in Figure 8. According to 32 submissions (80%), the feedback was presented fairly, or very clearly.

The third question was "*How helpful was the feedback received?*". Similarly to the second question, the students had a number of options to choose from (Figure 9). According to 28 of the submissions (70%), the feedback was fairly, or very helpful, while a 20% did not answer this question.

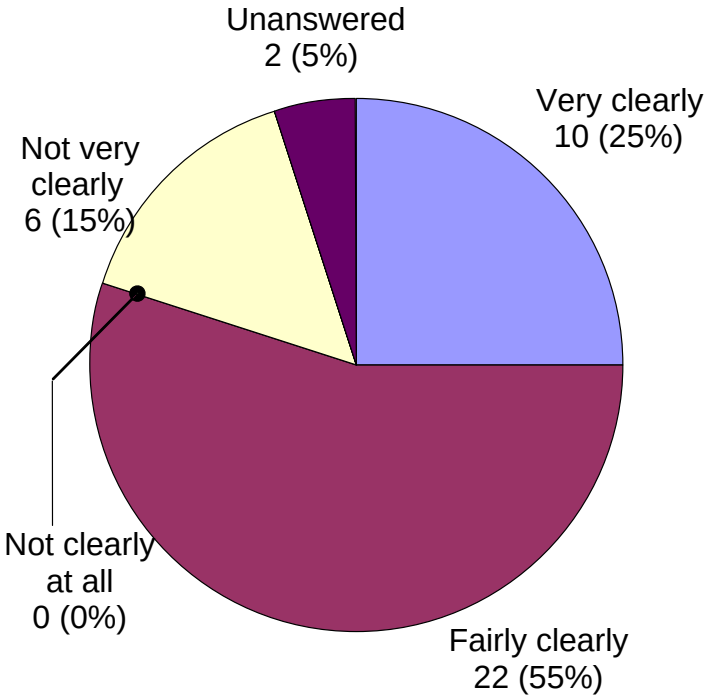

Figure 8: Answers to question "*How clearly was the feedback information presented?*".

The last question was "*What would you suggest to make the feedback mechanism better?*". A text area allowed the students to enter text of any length. Listing 3 displays the responses (14, since there was no response in the rest of the submissions) to this question. It is worth noting that the 5 students who responded purely positively in this question (cases 6, 7, 12, 13 14), were awarded high marks. The comments including constructive feedback touched mostly issues with the mechanism, that were known in advance. For instance, label matching (4, 8) and popup positioning (5, 10, 11) were not optimal. Additionally, some of the suggested defects were intended that way. For example, the message in the first comment is displayed whenever the current answer is almost identical to one of the specimen solutions stored in the gree, hence there is no useful feedback to be provided, although the message, could be clearer.

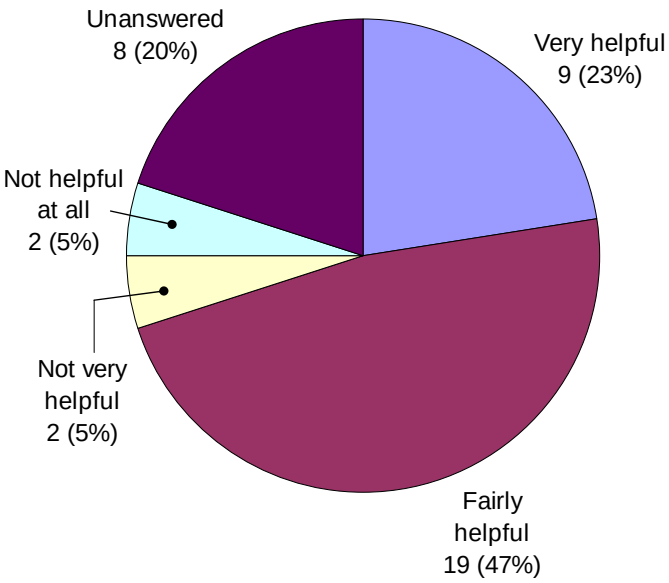

Figure 9: Answers to question "How helpful was the feedback received?".

Comment 9 is interesting because the tool was trying to direct the student towards the right answer - labels such as 1..* are never placed in the middle of a connector in UML - but the student was refusing to be helped!

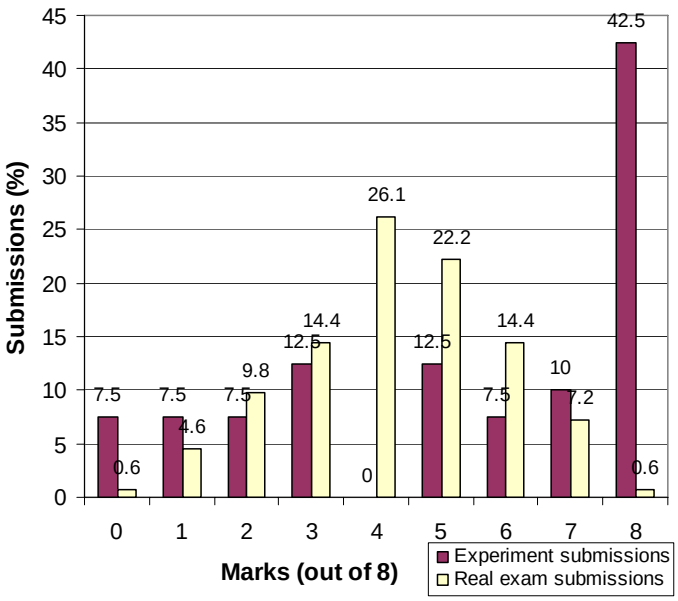

Figure 10: Final mark allocations.

The maximum marks available for the question were 8. 21 of the submissions (52.5%) were given an estimated mark, based on gree matching, between 7 and 8 marks. Figure 10 displays the marks the final submissions were awarded, compared against the marks awarded by a human marker for the real examination, a year earlier, when only one out of 153 students received full marks. Obviously, when using the feedback mechanism, the marks tend to be higher, while for the cases where the marks

were low, the feedback system was barely used and the question was probably abandoned half way through. Figure 11 shows that generally, the fewer the times the feedback mechanism was invoked, the lower the final mark. However, the lower right corner of this plot shows a number of high marks with relatively few hints, suggesting that the experiment prompted a number of students to do extra revision before using the tool.

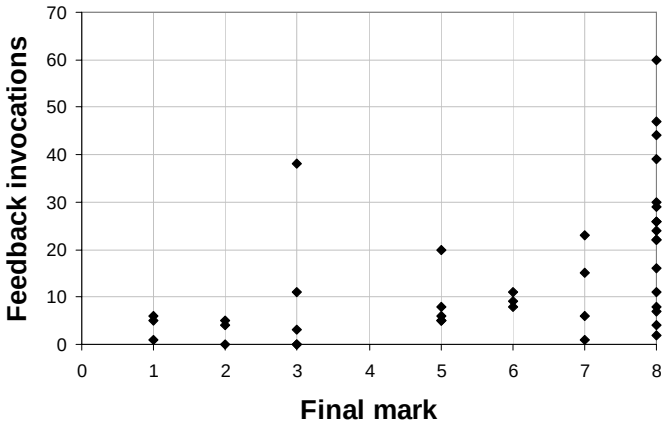

Figure 11: Marks – Feedback invocations correlation.

→	1.	Really quite good. Bug... 'No Feedback could be generated this time.' repeats. (8 marks, 4 hints)
↓	2.	Have it make sense. (3 marks, 3 hints)
→	3.	Better descriptions (2 marks, 4 hints)
→	4.	was very exact about names, it didnt recognise Reading material it wanted it without a space and Recommends > was told it shud be called recommends > (8 marks, 60 hints)
→	5.	sometimes they overlap which can be abit confusing/annoying. Maybe some kind of list of hints? like view next hint or something. Don't know if was intended but hints can just be used repeatedly to find the answer, but maybe that was the point? Also, I have no idea how many times I used hint.. It was lots. Very helpful anyway (7 marks, 15 hints)
↑	6.	Don't really know, its good at the moment and helped loads cheers :) (8 marks, 7 hints)
↑	7.	Nothing seems fine as it is (7 marks, 1 hint)
→	8.	More intuative suggestions, i.e. maybe more correct answers for it to choose from? The problem I had was that it would suggest that some of my correct aspects were incorrect and confuse me by telling me it was incorrect. (6 marks, 9 hints)
→	9.	Include more flexibility for labeling syntax. Such as allowing *..1 to be placed in the middle of the connection. (5 marks, 5 hints)
→	10.	Dont overlay the feedaback boxes (8 marks, 39 hints)
→	11.	Pop ups are a bit annoying, maybe have a feedback area and when feedback is clicked on area that needs changing is highlighted. (8 marks, 2 hints)
↑	12.	No need to improve (8 marks, 22 hints)
↑	13.	It's fine as it is (7 marks, 23 hints)

↑ 14. More questions to tackle with detailed feedback (5 marks, 5 hints)

Key: ↑ Purely positive comments
→ Constructive feedback
↓ Purely negative comments

Listing 3: Survey responses to the question "What would you suggest to make the feedback mechanism better?".

A similar question was set in the real exam in January 2007, and many more received high marks (7 or 8) compared to the previous year. This cannot be primarily due to the feedback tool, as less than a third of the students participated, in the experiment, but it suggests that the feedback tool may have have had a significant positive effect for some students. Figure 12 compares the marks awarded during the two examination runs.

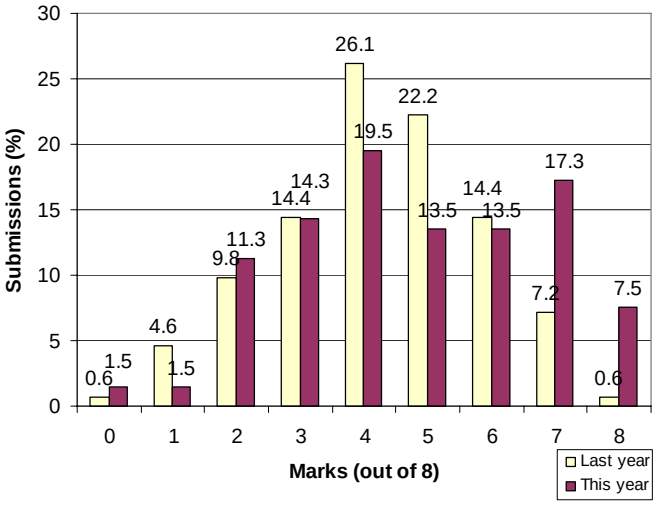

Figure 12: Comparison between the two examination runs.

To check that the estimated marks given by the feedback tool were reasonable, the matching mechanism was tested on a random sample of 48 out of 153 answers received for last year's examination. The automated marks were compared against the marks the human awarded. The results, shown in Figure 13 also indicate that the gree method has the potential to work effectively as a human marker's guide. Discrepancies are largely due to problems with label matching. For instance the

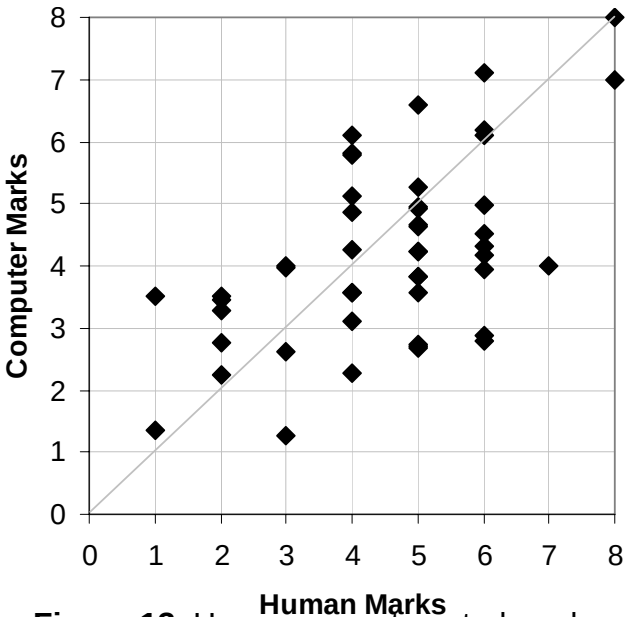

Figure 13: Human vs. automated marks.

labels “ReadingMaterial” and “Reading Material” are treated as different. Use of edit distancing and other techniques used in analysing text answers is required to address this issue.

Conclusions and further work

Grees, a special metaformat to represent structured answers including alternative parts and marking judgment information, independently of the knowledge domain, were introduced. They can be dynamically extended and in combination with a modularized matching mechanism, comparing and matching a submitted answer against a model answer is possible. The results can be expressed both visually and in terms of estimated marks.

The system was extended to include another mechanism that dynamically translates the matches into feedback combining explicit strings and visual information. A client application employing the whole system was deployed and 2nd year CS students were asked to answer a previous year examination question by drawing a UML diagram using it.

Although the trial group can be considered to be demanding given their exposure to computer systems, the experiment results proved to be clearly encouraging. The feedback mechanism was used several times per student and their final marks, compared to the ones from the examination the previous year, were significantly improved; in general, the more the feedback queries, the higher the final mark. According to the majority of the students, the feedback was at least “fairly helpful” and was presented at least “fairly clearly”. Some known problems, such as the weak label matching and the relative positions among the feedback popups, were pinpointed.

Since the reviewed draft of this paper, a second trial has been conducted, with first year AI students drawing Markov Chain diagrams. Although the type of diagram was quite different, the student feedback, both qualitative and quantitative, was very positive, and remarkably similar to that described above. This strongly reinforces the claim that a domain-independent representation can be used to give effective domain-specific formative feedback.

Future plans include testing the feedback system in other knowledge domains and even different types of constructed answers, such as mathematical expressions. Providing a user interface which allows markers to build and extend grees in an intuitive way remains an interesting challenge.

References

1. F. Batmaz and C.J. Hinde. A diagram drawing tool for semi-automatic assessment of conceptual database diagrams. In *10th International Computer Assisted Assessment Conference*, 2006.

2. J. Burtner, R. Rogge and L. Sumner. Formative assessment of a computer-aided analysis center: plan development and preliminary results. In *Frontiers in Education*, 2004.

3. P. Bocij and A. Greasley. Can computer-based testing achieve quality and efficiency in assessment? In *International Journal of Educational Technology*, 1999, vol 1.

4. C.A. Higgins and B. Bligh. Formative computer based assessment in diagram based domains. In *Innovation and Technology in Computer Science Education (ITiCSE) 2006*, 2006.

5. David J. Nicol and Debra Macfarlane-Dick. Formative assessment and self-regulated learning: a model and seven principles of good feedback practice. In *Studies in Higher Education*, Vol 31, No. 2, April 2006.

6. DR Sadler. Formative assessment: revisiting the territory. In *Assessment in education 5.1*, 1998.

7. John Sargeant, Mary McGee Wood and Stuart M. Anderson. A human-computer collaborative approach to the marking of free text answers. In *8th International Computer Assisted Assessment Conference*, 2004.

8. Neil Smith., Pete Thomas and Kevin Waugh. Interpreting imprecise diagrams. In *Diagrams 2004 Conference*, 2004.

9. Pete Thomas. Drawing diagrams in an online examination. In *8th International Computer Assisted Assessment Conference*, 2004.

10. Christos Tselonis, John Sargeant and Mary McGee Wood. Diagram matching for human-computer collaborative assessment. In *9th International Computer Assisted Assessment Conference*, 2005.

11. Athanasios Tsintsifas. *A framework for the Computer Based Assessment of Diagram Based Coursework.* PhD thesis, School of Computer Science and Information Technology, University of Nottingham, 2002.