Published in final edited form as:
Adv Health Sci Educ Theory Pract. 2008 December ; 13(5): 709–722. doi:10.1007/s10459-007-9081-3.

A Natural Language Intelligent Tutoring System for Training Pathologists - Implementation and Evaluation

Gilan M. El Saadawi, MD, PhD[1,2], **Eugene Tseytlin, MS**[1], **Elizabeth Legowski**[1], **Drazen Jukic, MD, PhD**[1,3,4], **Melissa Castine**[1], **Jeffrey Fine, MD, PhD**[4], **Robert Gormley, MD, PhD**[4], and **Rebecca S. Crowley, MD, MS**[1,4,5]

[1]Department of Biomedical Informatics, University of Pittsburgh School of Medicine

[2]Department of Health and Community Services, University of Pittsburgh School of Nursing

[3]Department of Dermatology, University of Pittsburgh School of Medicine

[4]Department of Pathology, University of Pittsburgh School of Medicine

[5]Intelligent Systems Program, University of Pittsburgh School of Arts and Sciences

Abstract

Introduction—We developed and evaluated a Natural Language Interface (NLI) for an Intelligent Tutoring System (ITS) in Diagnostic Pathology. The system teaches residents to examine pathologic slides and write accurate pathology reports while providing immediate feedback on errors they make in their slide review and diagnostic reports. Residents can ask for help at any point in the case, and will receive context-specific feedback.

Research Questions—We evaluated (1) the performance of our natural language system, (2) the effect of the system on learning (3) the effect of feedback timing on learning gains and (4) the effect of ReportTutor on performance to self-assessment correlations.

Methods—The study uses a crossover 2×2 factorial design. We recruited 20 subjects from 4 academic programs. Subjects were randomly assigned to one of the four conditions - two conditions for the immediate interface, and two for the delayed interface. An expert dermatopathologist created a reference standard and 2 board certified AP/CP pathology fellows manually coded the residents' assessment reports. Subjects were given the opportunity to self grade their performance and we used a survey to determine student response to both interfaces.

Results—Our results show a highly significant improvement in report writing after one tutoring session with 4-fold increase in the learning gains with both interfaces but no effect of feedback timing on performance gains. Residents who used the immediate feedback interface first experienced a feature learning gain that is correlated with the number of cases they viewed. There was no correlation between performance and self-assessment in either condition.

Keywords

Cognitive Modeling; *Dialogue-based interfaces*; *Health Sciences Education*; *Intelligent Tutoring Systems*; *Natural Language Processing*; *Pathology*

Corresponding Author: Dr. Rebecca Crowley UPMC Cancer Pavilion Suite 301, Room 307 5150 Centre Avenue Pittsburgh PA 15232 Ph: 412-623-1752 Fax: 412-623-2814 Email: crowleyrs@upmc.edu.

INTRODUCTION

Intelligent Tutoring Systems (ITS) are computer-based instructional systems with models of instructional content that specify what to teach, and teaching strategies that specify how to teach (Wenger, 1987). They make inferences about a student's mastery of topics or tasks in order to dynamically adapt the content or style of instruction. ITS support a style of learning best categorized as “learning by doing” - as students work on computer-based problems or simulations of real-world tasks, the ITS offers guidance, points out errors and organizes the curriculum to address the needs of that individual learner (Sleeman & Brown, 1982). In the last decade, ITS have moved out of the research laboratory and into classrooms and workplaces where some have been shown to be highly effective (Lajoie & Derry, 1993). Cognitive intelligent tutoring systems (CITS) incorporate domain-specific production rules that are based on a cognitive theory of skill acquisition(Anderson, Corbett, Koedinger, & Pelletier, 1995). Often, the intermediate cognitive steps are first identified using empirical methods such as cognitive task analysis. Some cognitive intelligent tutoring systems (CITS) use language based interfaces (Evens et al., 2001; Ros'e C., Litman D., Behembe D., Forbes K., & VanLehn K., 2003; VanLehn K., Jordan P. W., Ros'e C., & Wilson R., 2002). Both standard CITS [8] and those with language-based interfaces (Corbett, McLaughlin, & Scarpinatto, 2000; A. C. Graesser et al., 2004; Ros'e C. et al., 2003) have yielded successful evaluations with students.

For many years, researchers have been interested in developing dialogue-based educational systems that could interact with students by engaging them in conversation. It remains uncertain whether the use of such conversational interfaces will produce incremental gains in learning beyond the existing ITS methods(A. Graesser, Van Lahn, Rose, Jordan, & Harter, 2001) . However, some studies highlight the potential benefits of dialogue-based ITS. A meta-analysis of 65 studies done in a variety of instructional contexts by Cohen, Kulik, and Kulik 1982 (Cohen, Kulik, & Kulik, 1982) showed that human tutors produce remarkable learning gains - between 0.4 and 2.3 standard deviation units over classroom teaching. This was attributed to the “tutoring effect” and was true even when tutors used unstructured techniques and had little domain knowledge, suggesting that iterative probing may play an important part in enhanced learning. The effect of dialogue-based interaction on learning is further supported by studies that show that prompting students with little or no content in the tutoring interface is associated with an increase in the learning curves and that forcing them to generate words rather than simply reading them promotes subsequent recall of those words (Slamecka & Graf, 1978). There are at least two successful tutors that utilize conversational interface; Autotutor (A. C. Graesser et al., 2004) - a computer literacy tutor that simulates conversational dialogue and Atlas Andes and Why2 (Gertner & VanLehn, 2000; VanLehn K. et al., 2002) - a set of physics tutors that attempt to comprehend natural language and plan dialogue moves.

We have previously described SlideTutor (R.S. Crowley, Legowski, Medvedeva, & Tseytlin, 2005; R. S. Crowley et al., 2007; R.S. Crowley & Medvedeva, 2006) - an ITS that teaches visual classification problem solving based on a cognitive model of expertise in the domain of inflammatory diseases of skin. In the present study, we introduce a Natural Language Interface (NLI) that specifically analyzes and provides feedback on another important component of this task - the generation of a diagnostic report. Diagnostic reports are normally written in a paragraph format with standard terminology. Nevertheless, the words and concepts used are highly technical, and the skillful incorporation of these concepts is an important aspect of expertise in this task. Thus the nature of this task suggests a NLI could be beneficial to residents for learning how to write a report.

The style of feedback may be another variable in enhancing learning through conversational interfaces. Although studies of feedback found that immediate feedback is more effective than delayed feedback (Kulik & Kulik, 1988), a system that provides too much guidance may

interfere with the active nature of `learning by doing'. If immediate feedback through hints and bugs is readily available, students may not engage and attempt to solve the problem by themselves. Furthermore, if students come to rely on the system to help them in finding and fixing errors, they may learn less from these errors. Delayed feedback has its disadvantages also; studies have shown that delayed feedback may lead to unproductive floundering by the student resulting in failure to acquire tutored skills(Bangert-Drowns, Kulik, Kulik, & Morgan, 1991). So the question becomes: In what form and how much guidance should the ITS provide?

RESEARCH QUESTIONS

1. What is the performance of the natural language interface in determining the meaning of text input by students?
2. Is the use of ReportTutor associated with improved diagnostic report writing?
3. Does timing of feedback impact learning gains or acceptance of the system?
4. Is there a correlation between actual and perceived performance of subjects based on their self evaluation?

METHODS

System Design and Architecture

ReportTutor is designed to help residents learn how to analyze and report on melanoma cases - one of many domain areas covered by our tutoring system. Almost all academic departments of pathology now use the College of American Pathologists (CAP) checklists (“College of American Pathologists, Cancer Protocols and Checklists”) as a source of data elements (prognostic factors) which must be included in cancer reports such as those for melanoma biopsies and resections. The CAP cancer checklists are required for accreditation by the American College of Surgeons Commission on Cancer. The instructional goal of ReportTutor is to teach residents to correctly identify and document all relevant CAP prognostic factors in the diagnostic report.

ReportTutor is implemented as a client-server system, written in the Java programming language. The system architecture previously described (R.S. Crowley et al., 2005). Virtual slides for ReportTutor are scanned using a commercial robotic scanner (“Aperio Technologies”), and indexed in an Oracle database. Before use as ReportTutor cases, virtual slides must first be authored using our existing SlideAuthor system which is a Protégé (“Protégé. Retrieved From http://protege.stanford.edu/”) plugin. At the beginning of each case, ReportTutor creates a list of goals from the Protégé representation of the case created during the authoring process. The goal list is not displayed to the student, and goals are removed following completion. Individual goals may relate to actions only, actions combined with report features, or report features only.

Using the ReportTutor student interface (Figure 1-left) the student pans and zooms in a huge image file, simulating the use of a traditional microscope while the action detection component of the tutoring system monitors the actions of the student to determine whether the student has observed particular features or performed particular actions (such as measuring) correctly.

An interactive text entry area (Figure 1 - right) displays the major section headings for a generic pathology report. Some sections are static and pre-defined from the case representation (e.g.. clinical history, gross description) and some sections are empty, editable text-fields available for student to type into (e.g. final diagnosis, microscopic description, comment). The report feature detection component of the tutoring system monitors the actions of the student to determine whether the student has identified particular features by extracting known concepts

from the typed text that residents enter, and matching them to the domain ontology. Appropriate feedback is generated if some parts are incorrect or missing. In this way, residents must both correctly find and report on each required feature.

Study Design

To measure learning gains, we asked subjects to use the system for a total of 4 hours in the form of 2 working periods (one 2 hour working period for each interface), and then assessed performance before and after use of the system. To determine effect of feedback timing, we employed a crossover 2×2 factorial design to control for order of case presentation and interface use (Figure 2). The subjects were divided into 4 groups:

Group I: started with the Immediate Interface for cases 1 through 13 then used the Delayed Interface for cases 14 through 26.

Group II: started with the Immediate Interface for cases 14 through 26 then used the Delayed Interface for cases 1 through 13.

Group III: started with Delayed Interface for cases 1 through 13 then used the Immediate Interface for cases 14 through 26.

Group IV: started with the Delayed Interface for cases 14 through 26 then used the Immediate Interface for cases 1 through 13.

Subjects completed an assessment before and after each of the two working periods. During both working periods, we controlled for time on task, allowing a variable number of cases to be seen in the working period.

System Interfaces

To test the effect of feedback timing we used two interfaces that we developed for the ReportTutor backend. The ***immediate interface*** analyzes input text while it is being typed. This allows the tutor to provide the residents with positive and negative feedback immediately after each action in the interface. The feedback is provided using error messages, and also by continuously color-coding the text after residents type it into the interactive text entry area. When resident-entered text is parsed by the system and the concept is recognized, the concept text turns blue when it is correct, flashes if it is a feature that contains errors, or red if it is an incorrect attribute. If the text is not colored by the tutor it either means that this information is not important or the tutor failed to recognize the resident's wording. In addition to negative feedback, hints are also available. The resident can request a hint from the tutoring system as to what should be done next at any time. The resident has an indefinite number of attempts to submit his or her report. Only when the report is completely finished is the resident allowed to proceed to the next case. Using this interface, the resident is forced to write a complete and correct report and to correct mistakes as they appear.

The ***delayed interface*** analyzes text during the submission process. This interface doesn't provide any feedback while the report is being typed. The resident has three attempts to submit a report. When the report is submitted, the text is analyzed using the same matching engine as the previous interface, but instead of color coded text and flashing error messages, the user gets a list of mistakes as well as missed goals. The list can stay open while the resident addresses mistakes in the report. After a third submission, the resident can no longer change the report contents even if it is still incorrect or incomplete. At this point all of the hints for outstanding mistakes are provided. Residents using the delayed interface receive help only after each submission and some resident reports might still be incomplete or incorrect at the conclusion of the case.

System Cases

One hundred and fifty eight (158) de-identified reports were reviewed from 4 University Hospital archives and filtered to remove cases without residual melanoma yielding eighty-nine (89) cases that were obtained from the tissue bank. Only slides that were approved by the expert dermatopathologist as consistent with described features were scanned and used for the study. A total of 30 cases of melanoma and related lesions were included spanning different degrees of difficulty. Twenty six cases were used for the working periods on both interfaces and four cases were used for the assessment tests (Figure 2). Cases were randomly assigned to condition.

Reference Standard

As a reference standard for resident performance, we asked an expert dermatopathologist to view the same cases with the virtual microscope. The expert was the Chief Dermatopathologist at our institution - the most senior expert in this domain at our University Medical Center. The expert first dictated a diagnostic report that we captured as a text file. After all cases were dictated, he separately indicated the features, attribute names, and attribute values that ReportTutor is designed to detect (based on the CAP protocol data elements). These values were used as the “reference standard” against which we measured resident performance in identifying features, attributes and values.

Participants

The study was approved by the University of Pittsburgh Institutional Review Board (IRB Protocol # 0307023). Twenty subjects were recruited from four academic programs in Pennsylvania. Eight of the subjects were first year residents, eight were second year residents, and four were third year residents. Only seven residents had a previous dermatopathology rotation. All subjects were volunteers solicited by email and received a small honorarium for their participation.

Interface Training

The subjects were first trained on how to use both interfaces. Briefly, subjects watched a video demonstration of the demo training version of the system and then practiced using the system through a set of simple tasks.

Assessments

Subjects were given four interval assessments. All assessments were diagnostic cases in which residents used a virtual microscope and completed a diagnostic report, but without the assistance of any tutoring or feedback. Pretest and posttest for each working period were identical, and these cases were not used during the working session with the tutor. At the end of each assessment case, subjects were required to self grade their performance as a numerical score from 1-100.

Assessment Grading

Using discrete data form the reference standard, we manually coded each report for (1) whether each feature was present, (2) whether the attribute for the feature was present, and (3) whether the value for the attribute was correct. Two board-certified AP/CP pathology fellows (JF, RG) who were not involved with the experiment or system design manually coded the residents' assessment reports. The fellows were given guidelines with clear instructions on how to score the reports and were trained on a `practice training' report before they started scoring assessments. They used the reference standard described above as the correct answer set. To determine inter-rater reliability, we used a 20% overlap of reports (32 assessment tests) between graders.

The final score given to each report depended on presence of correct data (90% of score weight) and correct sequence or ordering (10%). Correct data was defined as identification of correct features, correct attributes and the correct values of these attributes. Points for correct ordering were given if the resident mentioned the most important features as identified by our expert in the beginning of the report.

Survey

After the completion of the final assessment, subjects were given a five-point Likert scale survey to determine their response to both interfaces, and which interface they preferred. The survey included items asking about user satisfaction on specific things about the system e.g. `Measuring tool was hard to use', `Virtual microscope viewer was easy to use', `Difficult to understand what the tutor did and did not recognize in my report' and *`Understood what I wrote'* as well as general questions about their overall impression e.g. `Enjoyable to use', `Provided relevant feedback', `Helped me learn how to write a report for melanoma cases', and `Made clear to me exactly what was wrong with my report'. Residents scored each item side-by-side for both interfaces to encourage direct comparison of the interfaces. At the end of the questionnaire some open-ended questions were added to rate both interfaces on various aspects of the system, and asked about suggestions to improve the system. These `open-ended questions' were not used for subsequent quantitative analysis.

NLI Performance Testing

We determined the accuracy of the NLI system in correctly recognizing text input for concepts, including features, attributes and values. We tested the system against pre-test 1 reports created by all 20 residents (40 total reports). Only reports written by residents before they used the tutoring system were included in the Natural Language Processing (NLP) performance metrics, because we reasoned that the system would `drive' residents to say things in a particular way. We used a small number of documents with a large number of individuals in this case because we specifically wanted to assure ourselves that the system would be accurate across the entire population of residents. We also used 24 clinical experts' reports that we had already collected from a previous study (R.S. Crowley, Tseytlin, & Jukic) to ensure that the language recognition capabilities of our system are not specific to any level of expertise.

These documents were then used to measure the performance of the system in detecting features and distinguishing correct from incorrect data elements. Reports processed by the system were coded to determine true positive (TP), false positive (FP), and false negative (FN) concepts. Precision (positive predictive value) indicates the proportion of concepts recognized correctly to the total number of recognized concepts (TP/TP+FP). Recall (sensitivity) indicates the proportion of concepts recognized correctly to the total number of concepts in text (TP/TP +FN). We determined frequencies of errors for all of ReportTutor's features and attributes, and computed precision and recall of the Report Feature detection system.

Analysis

Performance on pre-tests and post-tests was analyzed by MANOVA. We determined main effects and interactions for all assessments and interface condition. For performance-certainty correlations, slopes were computed using linear regression analysis, and were then compared by t-tests. The t-statistics were computed as the difference between the slopes divided by the standard error of the difference between the slopes. For the Likert scale portion of the survey, we compared interface conditions using student's t-test. All analyses were performed in SPSS.

RESULTS

Results of system evaluation

Table 1 shows performance metrics of the system. The precision is 0.90 and the recall is 0.84. These values are considered to reflect good performance when considered against current NLP systems.

Task metrics—There was a statistically significant difference in the number of cases viewed during the immediate interface working period (p =0.02); residents who started with the immediate interface viewed a smaller number of cases when compared with residents who started with the delayed interface. However, a comparable number of cases was viewed during the delayed interface in both conditions (p = 0.36). Figure 3 shows the mean number of cases seen by condition and Table 2 shows the average time spent (in minutes) per case with each interface by condition.

Inter-rater reliability for assessment grading—The overall inter-rater reliability was 94% agreement (kappa = 0.88). Considering features alone, the inter-rater reliability was 97% agreement (kappa = 0.94) and considering attributes alone, the inter-rater reliability was 94% (kappa= 0.86).

Learning outcomes

In both conditions, resident performance was significantly higher at post-test when compared to pre-test (Figure 4), showing an approximately four-fold gain. This was true for features (MANOVA, effect of test, F=320.7, p=<.001), attributes (MANOVA, effect of test, F=357.5 , p=<.001), and for both combined (MANOVA, effect of test, F=361.0, p=<.001).

As we expected, most of the learning gain was attained in the first working period, irrespective of the interface used. Residents in groups I and II, who used the immediate interface first experienced a feature learning gain that was correlated with the number of cases they viewed (r = 0.65, p = 0.04) - the more cases they saw with immediate feedback, the more they learned. This was not seen with the delayed interface (r= 0.30, p=0.399). There was no effect of post-graduate year on learning gains, which we have also shown in an ITS for teaching pathologic diagnosis [15,16].

Performance and Certainty Correlation

Although there was a statistically significant difference between pre-test and post-test self-scores (p<0.05) for all residents for both test 1 and test 2, analysis of self grading and performance showed that residents are relatively inaccurate in assessing their performance; there was no correlation between their self score and the actual test score given by the graders, and no improvement of the correlation after the use of the system, in either condition.

System Acceptance

Analysis of the Likert scale survey revealed that first year residents had a significantly higher total survey attitude score (73 ± 7.9) towards the immediate interface when compared with second (58 ± 14.3) and third year residents (64.4 ± 13.1) as shown in Figure 5 (t=3.16, p=0.007).

Although total attitude scores did not reveal a statistical difference in overall preference towards either interface, on average, residents felt that the delayed interface was easier to use and more flexible than the immediate interface (Figure 6)

DISCUSSION

The results of a previous study by Crowley et al., 2005[24], and this study show that the ReportTutor has a precision 0.90 and a recall 0.84. These values are considered to reflect good performance when considered against current NLI systems. There was a highly significant improvement in report writing after one tutoring session with 4-fold increase in the learning gains with both interfaces. This learning gain is equivalent to a previously reported study(R.S. Crowley et al., 2005; R. S. Crowley et al., 2007) in a different domain, where the diagnostic reasoning skills of residents showed a highly significant improvement after one tutoring session with SlideTutor. Retention tests used in that study showed that learning gains were undiminished after one week. In both studies, the pre-test and post-tests were identical, and did not contain cases seen in the tutoring session. The equivalency of these tests is important because it is often unclear what makes one case more or less difficult than another. Our demonstration of a strong increase in performance from pre-test to post-test thus cannot be attributed to differences in level of difficulty of non-equivalent tests. We have now observed highly significant learning gains in two different domains with two different task types - strengthening our conclusion that intelligent tutoring has significant potential as a new educational technology for the health sciences.

Despite a variety of opinions in the field about the relative benefits of one feedback timing over the other, this study does not demonstrate an effect of feedback timing on performance gains. Several studies have analyzed the pedagogic strategies of human tutors to ascertain what underlies their effectiveness (Fox, 1991; Leinhardt & Ohlsson, 1990). These studies suggest that human tutoring is effective because the tutor provides a give and take, offering guidance when help is needed, but holding back enough to encourage autonomy. Multiple studies show that encountering obstacles and working through them can be an important step in the learning process (Chi M, Siler A., Jeong H., Yamauchi T., & Lavancher C, 2001; Ohlsson & Rees, 1991). However, there are potential drawbacks to this approach. Residents trying to solve problems can expend a lot of time and effort engaging in parts of solution space that are not productive.

Interestingly, residents in groups I and II, who used the immediate feedback interface first experienced a feature learning gain that is correlated with the number of cases they viewed - the more cases they saw with the immediate tutor, the more they learned. Overall, residents in this group saw fewer cases. This might suggest that the immediate interface is more efficient than the delayed interface, leading to equivalent gains with fewer cases. One implication of these results is that immediate feedback may be helpful in domains where it is necessary to compensate for a restricted case base.

Our final research question focused on potential meta-cognitive differences between the two types of feedback. An important step in development of expertise is to match certainty to performance or to link the subjective and objective indices of knowledge i.e. engage in the feeling of knowing (FOK)(Nelson, 1984). When practitioners are uncertain about a diagnosis, they can seek consultation from an expert, because being overly certain about a diagnosis that turns out to be wrong could be significantly harmful. In a previous study of SlideTutor (R.S. Crowley et al., 2005), we observed an improvement in self-assessments when residents used knowledge-centric representation but not with case-centric representation. This effect was not observed in this study. Interestingly, we found no correlation between resident's self score and the actual test score given by the graders, and no improvement of the correlation after the use of the system.

Why did residents who used this system fail to improve their feeling of knowing (FOK)? The answer may lie in the problem representation used by the interface. Our other tutoring system

relies heavily on a visual representation of the problem space. For example, the knowledge representation in SlideTutor enables the resident to see the whole problem space as a diagnostic tree. Residents see the effect of subtle differences in diagnosis across all cases. In contrast, ReportTutor uses an entirely text interface where residents cannot easily compare their performances to a standard across many cases.

FUTURE WORK

This study has sparked our interest to conduct further studies to examine the impact of meta-cognitive tutoring on skills. In a follow-on study, we are attempting to de-bias the resident confidence levels and use scaffolds to help improve their feeling of knowing (FOK) and judgment of learning (JOL) (Koriat, 1997). Studies have indicated that FOK is a predictor of learning gains and future performance (Carroll & Nelson, 1993). Since our ITS adequately satisfy all the required characteristics as defined by Azevedo (Azevedo, 2002; Azevedo & Lajoie, 1998) we plan to use it as a meta-cognitive tutoring layer to test whether this kind of tutoring helps residents to develop a more highly correlated judgment of their own performance.

ACKNOWLEDGMENTS

Work on ReportTutor is supported by a grant from the National Cancer Institute (R25 CA101959). This work was conducted using the Protégé resource, which is supported by grant LM007885 from the United States National Library of Medicine. We gratefully acknowledge the contribution of the SPECIALIST NLP tools provided the National Library of Medicine. We thank Olga Medvedeva for her expert technical help with this project, and Lucy Cafeo and Maria Bond for editorial assistance.

REFERENCES

Anderson JR, Corbett AT, Koedinger KR, Pelletier R. Cognitive tutors: lessons learned. Journal of the Learning Sciences 1995;4(2):167–207.

Aperio Technologies. from http://www.aperio.com/

Azevedo R. Beyond intelligent tutoring systems: Using computers as METAcognitive tools to enhance learning? Instructional Science 2002;30:31–45.

Azevedo R, Lajoie SP. The cognitive basis for the design of a mammography interpretation tutor. International Journal of Artificial Intelligence in Education 1998;9:32–44.

Bangert-Drowns RL, Kulik CC, Kulik JA, Morgan MT. The instructional effect of feedback in test-like events. Review of Educational Research 1991;61(2)

Carroll M, Nelson TO. Effect of Overlearning on the Feeling of Knowing Is More Detectable in Within-Subject than in Between-subject Designs. The American Journal of Psychology 1993;106(2):227–235. [PubMed: 8338189]

Chi M, Siler A, Jeong H, Yamauchi T, Lavancher C. Learning from human tutoring. Cognitive Science 2001;25:471–553.

Cohen PA, Kulik JA, Kulik CC. Educational Outcomes of Tutoring: A Meta-analysis of Findings. American Educational Research Journal 1982;19(2):237–248.

College of American Pathologists. Cancer Protocols and Checklists. from http://www.cap.org/apps/docs/cancer_protocols/protocols_index.html

Corbett A, McLaughlin M, Scarpinatto KC. Modeling student knowledge: cognitive tutors in high school and college. User Modeling and User-Adapted Interaction 2000;10:81–108.

Crowley, RS.; Legowski, E.; Medvedeva, O.; Tseytlin, E. An ITS for medical classifcation problem-solving: effects of tutoring and representations; Paper presented at the Proceedings of the 12th International Conference on Artificial Intelligence (AIED05); Amsterdam, the Netherlands. 2005.

Crowley RS, Legowski E, Medvedeva O, Tseytlin E, Roh E, Jukic D. Evaluation of an intelligent tutoring system in pathology: effects of external representation on performance gains, metacognition, and acceptance. J Am Med Inform Assoc 2007;14(2):182–190. [PubMed: 17213494]

Crowley RS, Medvedeva O. An intelligent tutoring system for visual classification problem solving. Artificial Intelligence in Medicine 2006;36:85–117. [PubMed: 16098717]

Crowley, RS.; Tseytlin, E.; Jukic, D. ReportTutor - an intelligent tutoring system that uses a natural language interface; Paper presented at the AMIA 2005 Symposium Proceedings; Washington, DC.

Evens, MW.; Brandle, S.; Change, RC.; Freedman, R.; Glass, M.; Lee, YH., et al. CIRCSIM-Tutor: an intelligent tutoring system using natural language dialogue; Paper presented at the Twelth Midwest AI and Cognitive Science Conference (MAICS); Oxford, OH. 2001.

Fox, B. Cognitive and Interactional Aspects of Correction in Tutoring; Paper presented at the Teaching Knowledge and Intelligent tutoring; Norwood, NJ: Ablex; 1991.

Gertner, A.; VanLehn, K. ANDES: a coached problem solving environment for physics. In: Gautheir, G.; Frasson, C.; VanLehn, K., editors. Intelligent Tutoring Systems: 5th International Conference (ITS-2000); Berlin: Springer; 2000. p. 131-142.

Graesser A, Van Lahn K, Rose C, Jordan P, Harter D. Intelligent tutoring systems with conversational dialogue. AI Magazine 2001;v22(i4):39.(13)

Graesser AC, Lu S, Jackson GT, Mitchell HH, Ventura M, Olney A, et al. AutoTutor: a tutor with dialogue in natural language. Behav Res Methods Instrum Comput 2004;36(2):180–192. [PubMed: 15354683]

Koriat A. Monitoring one's own knowledge during study: A cue-utilization approach to judgments of learning. Journal of Experimental Psychology 1997;126(4):349–370.

Kulik JA, Kulik CC. Timing of feedback and verbal learning. Review of Educational Research 1988;58 (1):79–97.

Lajoie, S.; Derry, SJ. Computers as cognitive tools. Lawrence Erlbaum Associates; Hillsdale, N.J.: 1993.

Leinhardt G, Ohlsson S. Tutorials on the structure of tutoring from teachers. Journal of Artificial Intelligence in Education 1990:238–256.

Nelson TO. A comparison of current measures of the accuracy of feeling-of-knowing predictions. Psychol Bull 1984;95(1):109–133. [PubMed: 6544431]

Ohlsson S, Rees E. The function of conceptual understanding in the learning of arithmatic procedures. Cognition and Instruction 1991;8:103–179.

Protégé. Retrieved From http://protege.stanford.edu/

Ros'e, C.; Litman, D.; Behembe, D.; Forbes, K.; VanLehn, K. A comparison of tutor and student behavior in speech versus text based tutoring. Paper presented at the Proc. HLT/NAACL Workshop: Building Educationan Applications Using NLP; 2003.

Slamecka NJ, Graf P. The generation effect. Delinneation of a phenomenon. Journal of experimental Psychology. Human Learning and Memory 1978;4:592–604.

Sleeman, D.; Brown, JS. Intelligent Tutoring Systems. Academic Press; London: 1982.

VanLehn K, Jordan PW, Ros'e C, Wilson R. The architecture of Why-2 atlas: A coach for qualitative physics essay writing. Procedure of Intelligent Tutoring Systems. 2002

Wenger, E. Artificial Intelligence and Tutoring Systems - Computational and Cognitive Approaches to the Communication of Knowledge. Kaufmann Publishers; Los Altos, CA: 1987.

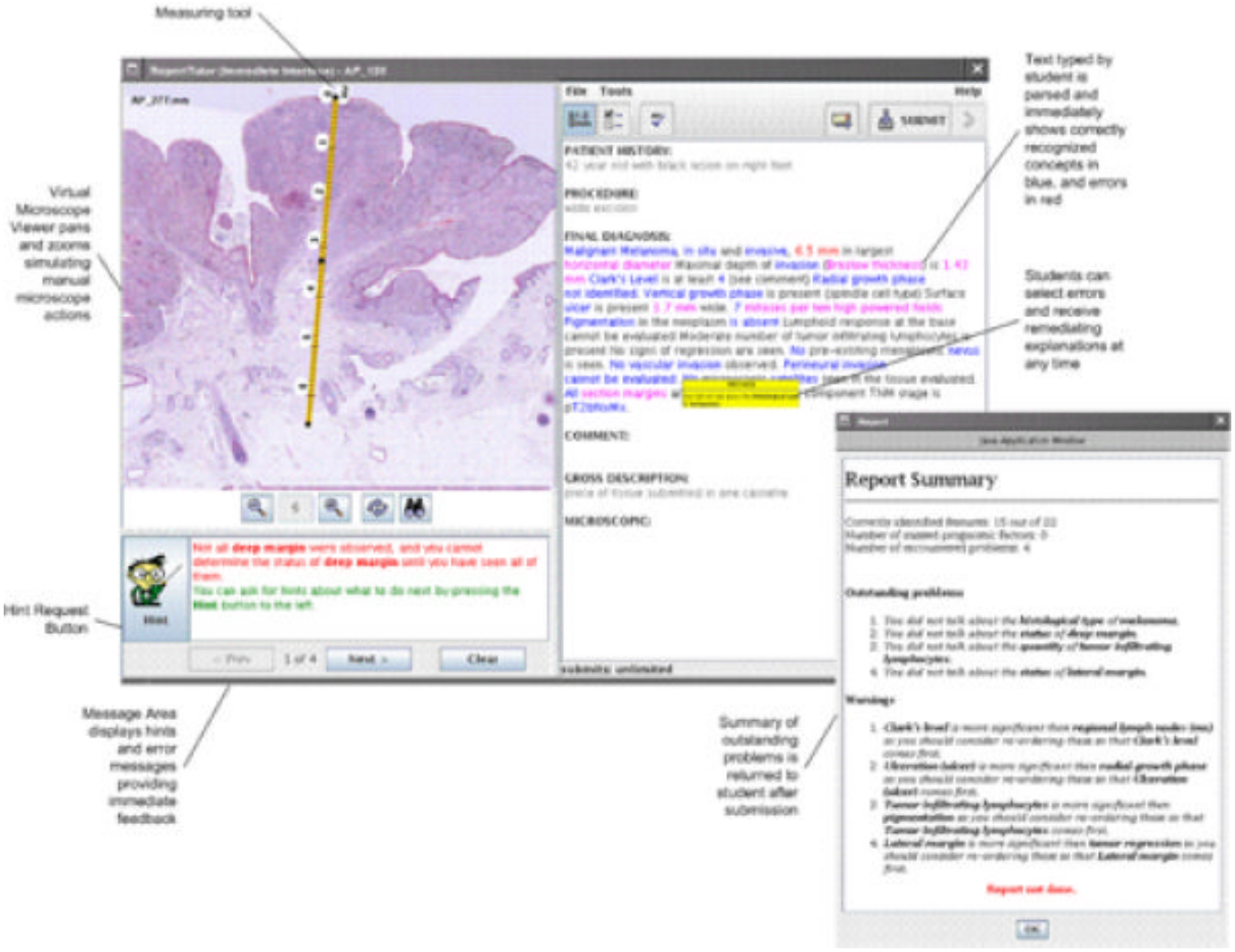
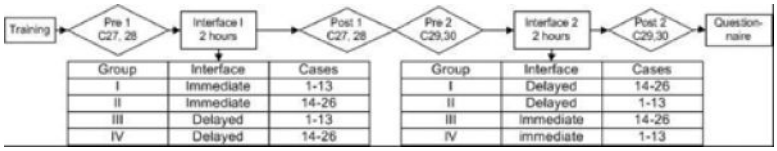


Figure 1.
ReportTutor NLP interface (approx pg. 6)



Group	Interface	Cases
I	Immediate	1-13
II	Immediate	14-26
III	Delayed	1-13
IV	Delayed	14-26

Group	Interface	Cases
I	Delayed	14-26
II	Delayed	1-13
III	Immediate	14-26
IV	immediate	1-13

Figure 2.
Study design (approx pg. 8)

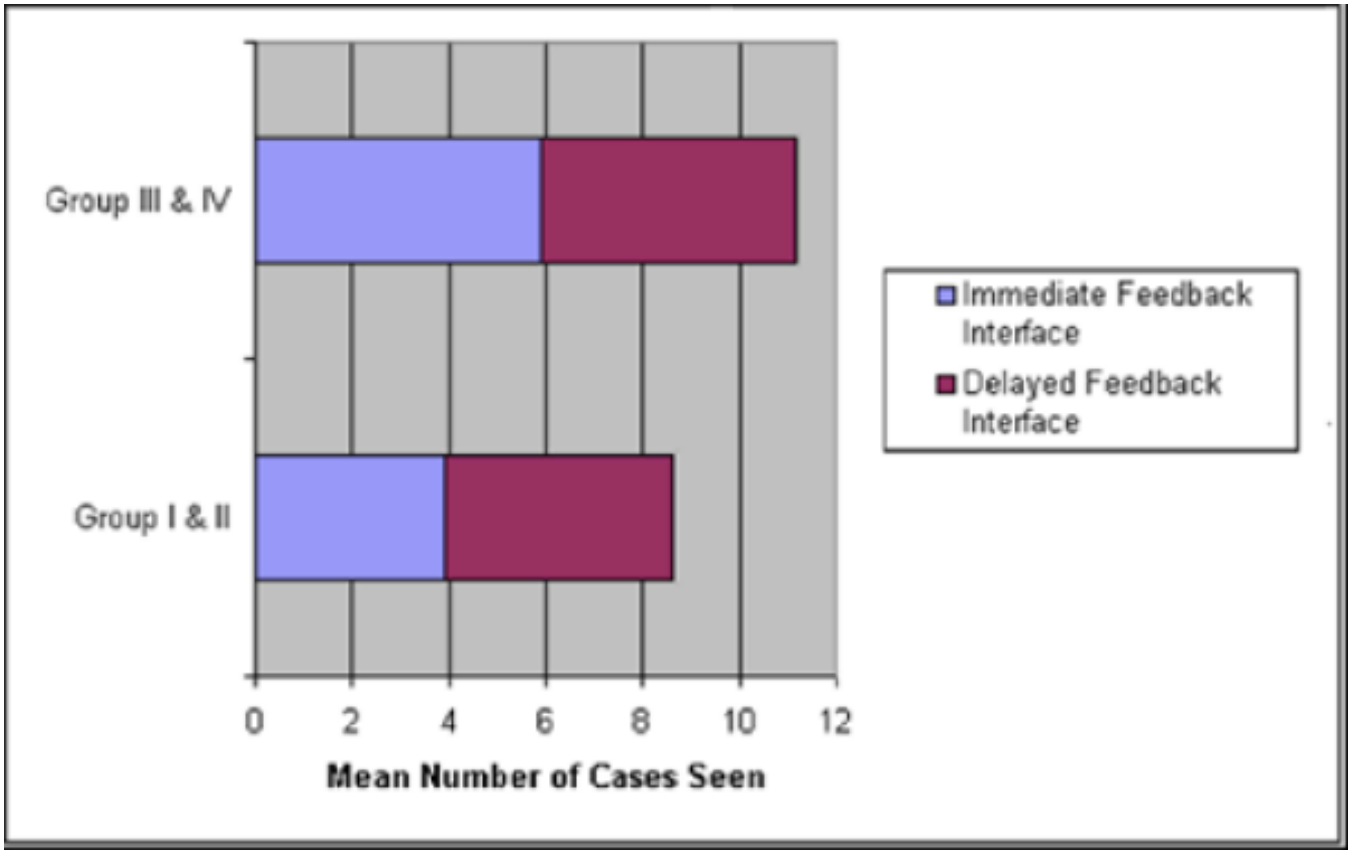


Figure 3.
Mean number of cases seen by condition (approx pg 14)

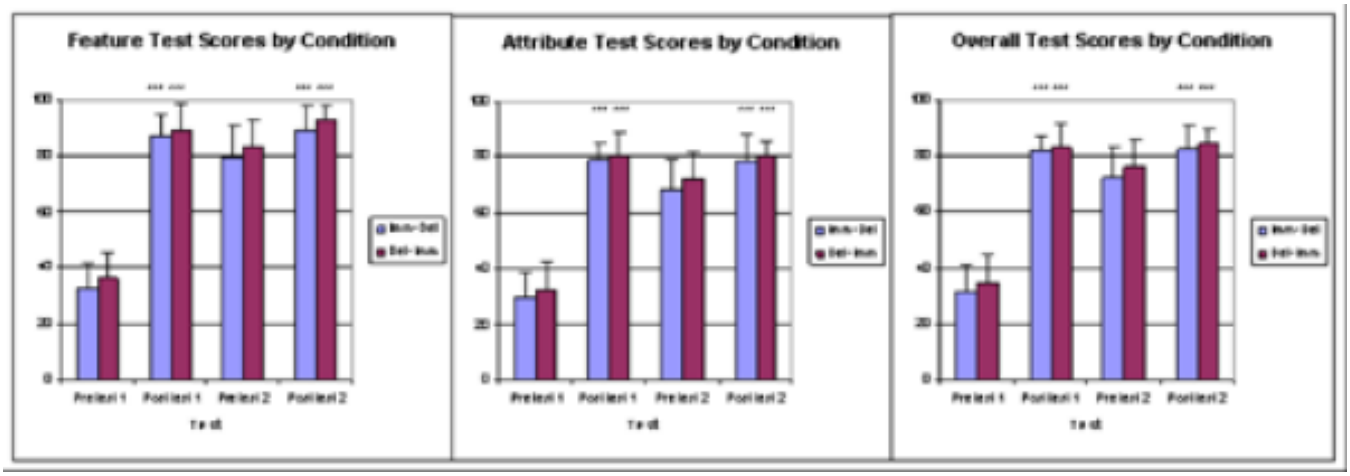


Figure 4.
Learning gains for features, attributes, and overall combined (approx pg. 15)

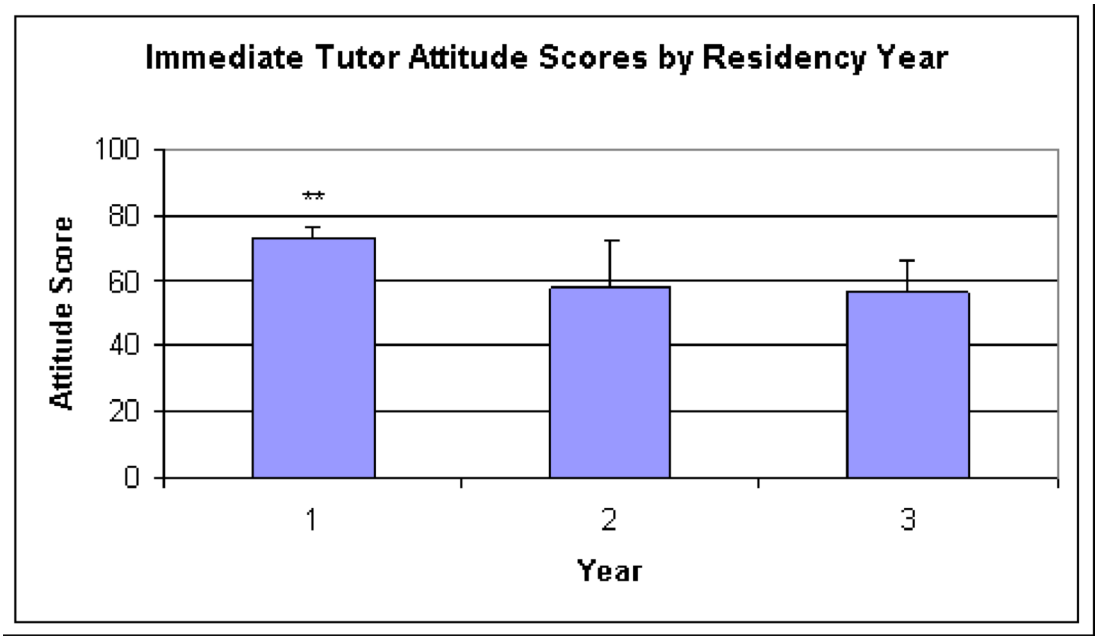


Figure 5.
Higher attitude score towards immediate interface by first year residents (approx pg. 16)

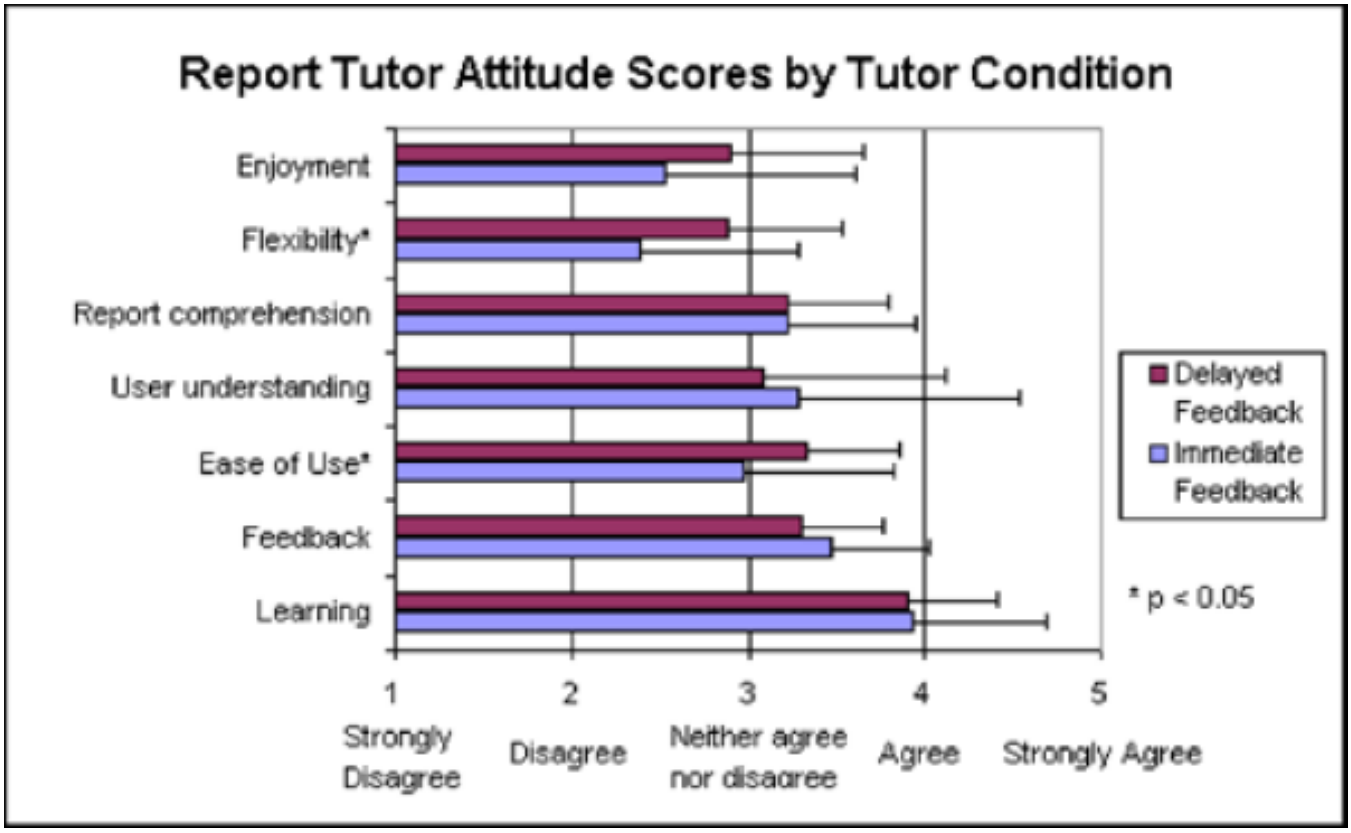


Figure 6.
Attitude score by condition (approx pg. 16)

Table 1

NLP performance metrics

	Expert Reports	Resident Reports	All Reports
TP	592 (74.09%)	450 (81.52%)	1042 (77.13%)
FN	125 (15.64%)	72 (13.04%)	197 (14.58%)
FP	82 (10.26%)	30 (5.43%)	112 (8.29%)
Sensitivity (Recall)	0.82	0.86	0.84
PPV (Precision)	0.94	0.88	0.9

Table 2

Average time spent (in minutes) per case with each interface by condition

Condition	Immediate Feedback		Delayed Feedback	
	Mean	**SD**	**Mean**	**SD**
Immediate-Delayed	35.2	14.67	26.31	6.44
Delayed-Immediate	22.73	8.13	24.03	6.82