Generalizing Automated Detection of the Robustness of Student Learning in an Intelligent Tutor for Genetics

Ryan S.J.d. Baker

Teachers College, Columbia University

Albert T. Corbett

Carnegie Mellon University

Sujith M. Gowda

Worcester Polytechnic Institute

Corresponding author: Ryan S.J.d. Baker

Contact email: baker2@exchange.tc.columbia.edu

Address: Department of Human Development, Teachers College, Columbia University, 525 W. 120th Street, New York, NY 10522 USA.

Telephone: (+1) 212- 678-8329

Fax: (+1) 212-678-3837

Abstract

Recently, there has been growing emphasis on supporting robust learning within intelligent tutoring systems, assessed by measures such as transfer to related skills, preparation for future learning, and longer-term retention. It has been shown that different pedagogical strategies promote robust learning to different degrees. However, the student modeling methods embedded within intelligent tutoring systems remain focused on assessing basic skill learning rather than robust learning.

Recent work has proposed models, developed using educational data mining, that infer whether students are acquiring learning that transfers to related skills, and prepares the student for future learning (PFL). In this earlier work, evidence was presented that these models achieve superior prediction of robust learning to what can be achieved by traditional methods for student modeling.

However, using these models to drive intervention by educational software depends on evidence that these models remain effective within new populations. To this end, we analyze the degree to which these detectors remain accurate for an entirely new population of high school students. We find limited evidence of degradation for transfer. More degradation is seen for PFL; this degradation appears to occur in part because it is generally more difficult to infer this construct within the new population.

Keywords: Robust learning, preparation for future learning, transfer, student modeling, intelligent tutoring system

Introduction

Increasingly, it is thought desirable that students acquire what is termed “robust” knowledge (Koedinger et al., 2012): knowledge grounded in conceptual domain knowledge (Craig, VanLehn, & Chi, 2008), which transfers more readily to related problem situations (Singley & Anderson, 1989; Fong & Nisbett, 1991), is retained by students over time (Schmidt & Bjork, 1992; Bahrick, Bahrick, Bahrick, & Bahrick, 1993), and prepares students for more efficient or more effective future learning (Bransford & Schwartz, 1999; Schwartz & Martin, 2004). One of the well-documented risks in problem solving across STEM domains is that students can develop superficial knowledge that fails these tests of robust learning. In particular, when students are not well-prepared for problem solving, they can develop problem solving knowledge which focuses on surface elements in problem situations, formal representations, and features of the learning environment itself (Chi, Feltovich, & Glaser, 1981; Rittle-Johnson & Siegler, 1998).

In line with this shift in perspective, over the past 15 years there has been a growing effort by intelligent tutoring system (ITS) developers and developers of other intelligent learning environments (ILEs) to develop interventions explicitly designed to increase the robustness of student learning. One general theme has been to improve the effectiveness of tutor feedback in supporting deep understanding, e.g., through natural language tutorial dialogues (Graesser et al., 2004; Katz, Connelly, & Wilson, 2007), through enhanced student interactivity with graphical feedback (Corbett & Trask, 2000; Butcher, 2010), or through focusing feedback on domain-independent strategies (Chi & VanLehn, 2007). A second major approach has focused on incorporating student-explanations into ITSs, asking students to explain their actions in problem solving (Aleven & Koedinger, 2002), or to explain worked examples of problem solutions (Corbett, et al. 2011; Hausmann & VanLehn, 2007; McLaren, Lim, & Koedinger, 2008;

Schwonke, Renkl, Krieg, Wittwer, Aleven, & Salden, 2009), towards supporting students in monitoring their understanding. Other efforts have focused on training meta-cognitive skills, such as the skill of using a tutoring system's corrective and explanatory feedback effectively (Aleven, McLaren, Roll, Koedinger, 2006; Roll, Aleven, McLaren, Koedinger, 2007), and providing meta-cognitive feedback on students' skill at self-regulated learning (Chin et al., 2010; Tan & Biswas, 2006).

The advent of interventions that can support the development of robust learning raises the question of whether another major benefit of intelligent tutors and AIED technologies can be leveraged: individualization. Individualization is a major goal of ITS and AIED systems (cf. McCalla, 1992; VanLehn, 2006), driven by models of students' latent knowledge (cf. Martin & VanLehn, 1995; Corbett & Anderson, 1995; Shute, 1995). Individualization based on student knowledge has had substantial benefits for learners. For instance, Corbett (2001) demonstrated that Bayesian student modeling can be used to more efficiently distribute problem solving practice in an ITS, leading to a large gain in mean post-test accuracy with only a small additional cost in total time on task, compared to a fixed curriculum. Bayesian student modeling has also been successfully used to monitor student explanations of worked examples in ITSs (Conati, Gertner, & VanLehn, 2002; Salden, Koedinger, Renkl, Aleven, & McLaren, 2010).

Efforts to individualize learning environments rely on accurate student modeling. The efforts listed above have leveraged models of student knowledge which can successfully infer the probability that a student knows a specific skill from the student's history of correct responses and non-correct responses (e.g. errors and hint requests) for that skill up until that time (cf. Corbett & Anderson, 1995; Martin & VanLehn, 1995; Shute, 1995; Pavlik, Cen, & Koedinger, 2009). In recent years, the debate about how to best model student knowledge has continued, with an increasing number of explicit comparisons of models' ability to predict future performance within the tutoring software studied (cf. Pavlik et al., 2009; Gong, Beck, & Heffernan, 2010; Wang & Heffernan, 2011; Pardos et al., 2011).

While these student modeling approaches have been successful at predicting immediate problem-solving performance and improving performance on those tests, less attention has been paid to modeling the robustness of student learning. Several studies have shown that Bayesian student modeling can accurately predict immediate post-test performance on the same problem-solving skills studied in a tutor (e.g., Corbett & Anderson, 1995; Shute, 1995; Baker et al., 2010; Corbett, Maclaren, Kauffman, Wagner, & Jones, 2010; Pardos et al., 2011), a very limited form of transfer. But student models in intelligent tutoring systems have typically not attempted to go beyond this point in modeling whether learning is robust. Relatedly, some results suggest that Bayesian student modeling can be insensitive to differences in students' depth of understanding. For example, Corbett and Anderson (1995) report that while Bayesian student modeling achieved high correlation to student post-test performance in the APT Lisp Tutor, it overestimated average student post-test performance by 5-10%. Tellingly, Corbett & Bhatnagar (1997) found that the extent to which the student model overestimates student test performance is inversely correlated with the each student's initial declarative knowledge. In another APT Lisp Tutor study (Corbett & Trask, 2000), two groups of students worked to cognitive mastery levels with conventional and enhanced feedback related to a difficult topic. While students in the two groups worked to the same nominal cognitive mastery criterion, students in the enhanced feedback condition scored reliably better on the post-test, again suggesting that this type of student modeling may be partially insensitive to differences in deep understanding.

Some steps in the direction of modeling the robustness of learning in intelligent tutoring systems have been taken. For example, Jastrzembski, Gluck, and Gunzelmann (2006) predict not just post-test performance, but also how long knowledge will be retained after learning, within an ITS teaching flight skills. Another step in this direction is to assess the transfer of skill within the learning system. Much of this work has taken the form of modeling inter-connections between skills during learning (cf. Martin & VanLehn, 1995) or online testing (Desmarais, Meshkinfam, & Gagnon, 2006), or in using inter-connections between skills to revise skill models (Pavlik, Cen, Wu, & Koedinger, 2008). Additional, computational modeling has analyzed the mechanisms leading to accelerated future learning within a learning system (Li, Cohen, & Koedinger, 2010).

Building on this work, recent work has used data mining to develop models that can automatically detect whether student knowledge will transfer to related skills outside of the tutoring system, and whether students are prepared for future learning outside of the tutoring system. The difference between transfer and PFL is whether students have the ability to directly apply their existing knowledge in novel situations or in new fashions (transfer), versus whether students can acquire new knowledge more quickly or effectively from future instruction, using their existing knowledge (PFL). If models are developed that accomplish these goals – predicting from in-tutor behavior whether a student will be able to successfully transfer their knowledge out of the tutor to different skills and situations, and whether a student will be prepared for future learning outside of the tutor – then these models could be used to identify students who may be developing superficial knowledge in problem solving and in selecting interventions designed to improve the robustness of student learning. Students who are already on the road to robust learning could continue with existing activities, while students unlikely to achieve robust learning could receive interventions.

In this earlier work, robust learning detectors (for both transfer and preparation for future learning) were developed for a population of undergraduate students using a Cognitive Tutor in the domain of Genetics problem solving (Corbett et al., 2010). These detectors were generated by engineering complex features related to students' motivation and meta-cognition, and creating a model to predict transfer/PFL from these features. They were assessed using cross-validation at the student level (e.g. the detectors were repeatedly developed using data from one group of students and tested on other students). The detectors were found to be better than traditional student modeling methods for predicting both Transfer and PFL. In this paper, we study how well these detectors of Transfer and PFL generalize at the population level, studying the degree to which they transfer to a new group of students; specifically, a younger group of high school students using the same tutor software.

In addition to examining the models' degree of generalization, we also analyze the specific student behaviors that are associated with robust learning in each population, towards increasing understanding of the conditions under which robust learning occurs in interactive learning systems of this type.

Learning System

Cognitive Tutors are a type of interactive learning environment which use cognitive modeling and artificial intelligence to model student learning, in turn using the model of student learning to adapt to individual differences in student knowledge and learning (Koedinger & Corbett, 2006). Cognitive Tutor curricula combine conceptual instruction delivered by a teacher with computer-based learning where each student works one-on-one with a Cognitive Tutoring system that chooses exercises and feedback based on a running model of which skills the student possesses (Corbett & Anderson, 1995).

Within a Cognitive Tutor, as the student works through a set of problems, Bayesian Knowledge Tracing (Corbett & Anderson, 1995) is used to determine how well the student is learning component skills, calculating the probability that the student knows each skill based on that student's history of responses within the tutor. Using these estimates of student knowledge, the tutoring system gives each student problems that are relevant to the skills which he or she needs to learn, continuing to provide problems until the student reaches mastery (e.g. 95% probability of knowing each skill) on all skills relevant to a given curricular area.

Within this paper, we study robust learning in the context of the Genetics Cognitive Tutor (Corbett et al., 2010). This tutor consists of 19 modules that support problem solving across a wide range of topics in genetics (Mendelian transmission, pedigree analysis, gene mapping, gene regulation and population genetics). Various subsets of the 19 modules have been piloted at 15 universities in North America. This study focuses on a tutor module that employs a gene mapping technique called a *three-factor cross* (3FC). The tutor interface for this reasoning task is displayed in Figure 1. The 3FC technique is used to determine both the order of three genes, (F, G, and H in this example), which lie on one chromosome, and to find the relative distances between the pairs of genes. In this technique two organisms are crossed (two fruit flies in the example) and the resulting distribution of offspring phenotypes is analyzed to infer the arrangement of the three genes on the chromosome. In Figure 1 the student has almost finished the problem. The student has summed the number of offspring in each of four phenotype groups that appear in the offspring table, and has categorized each group (as "parental" "single crossover" during meiosis, or "double crossover" during meiosis). The student has compared the phenotype patterns in the offspring groups, to identify the middle of the three genes and entered a gene sequence below the table. Finally, in the lower right the student has calculated the crossover frequency between two of the genes, G and H, and the distance between the two genes. The student will perform the last two steps for the other two gene pairs.

(Place Figure 1 approximately here)

Robust Learning Measures

The robustness of student learning was measured through two tests: a transfer test, and a preparation for future learning (PFL) test. A standard pre-test and post-test, measuring the exact skills studied in the tutor, were also given.

The transfer test consisted of two problems. The first problem was a three-factor cross task in which double crossovers were so improbable that the double-crossover offspring group was missing. This is a "gap filling" transfer task (cf. VanLehn et al., 1992). The problem is solvable and most of the students' problem-solving knowledge directly applies; the task examines whether students can draw on their understanding of that problem-solving knowledge to fill in the "gap" that results from the missing offspring group. The second problem examines whether students can extend their understanding of crossovers and crossover notation from three genes to four genes. In this problem students were given a parental genotype with four genes and asked to identify how many crossovers had occurred in various offspring groups (based on phenotype structure rather than relative frequency) and to identify all the offspring groups in which a specific crossover had occurred. Students completed this transfer test following the problem-solving post-test at the end of session 2.

It is worth noting that the form of transfer represented by these problems can be seen as different from simply transferring knowledge to an isomorphic problem (cf. Gick & Holyoak, 1987). However, transfer problems of the more complex nature seen here, requiring some reasoning beyond simply transfer of skill, are frequently also seen in research on robust learning in interactive learning software (cf. Aleven & Koedinger, 2002; Atkinson, 2002; Mathan & Koedinger, 2005; Hausmann & VanLehn, 2007), and may represent a deeper test of the robustness of knowledge than an isomorphic problem. Interestingly, this more complex type of transfer problem is sometimes termed “far transfer”, but it is not yet clear whether it is more difficult for students to modify their knowledge to accomplish a related task (the type of transfer seen here), or whether it is more difficult for them to realize that their existing knowledge applies in a different context (the type of transfer studied in Gick & Holyoak, 1987).

In the preparation for future learning test, students were asked to solve parts of a four-factor cross problem. The reasoning is related to solving a three-factor cross problem, but sufficiently more complicated that a student could not be expected to invent a solution method by direct transfer, and certainly not in a short period of time. Consequently, this PFL test presented a 2.5-page description of the reasoning in a four-factor cross experiment, then asked students to solve some elements of a four-factor cross problem: identifying the middle genes, identifying all the offspring groups with a crossover between two specific genes and to find the map distance between those two genes.

Previous Models

In Baker, Gowda, & Corbett (2011a, 2011b), we presented models that can predict student transfer and preparation for future learning. These models were developed using data from 72 college students enrolled in biology courses at Carnegie Mellon University, who used the Genetics Cognitive Tutor for two hours apiece. The students used the Cognitive Tutor software for two hours, completing a total of 22,885 problem solving attempts across a total of 10,966 problem steps in the tutor.

Feature Engineering

The first step of our process of developing models of robust learning was to engineer a set of features based on a combination of theory and prior work detecting related behaviors. We tested a set of 18 features, represented as a set of 9 core features and 9 related features. Features 1-5 and their related features focus on student interactions with the tutor's hints and feedback. Features 6-8 and their related features focus on the student's problem solving actions. The 9th feature involves the dynamics of the student's learning, moment by moment.

1. Help avoidance (Aleven et al., 2006), not requesting help on poorly known skills (on the student's first attempt at a specific problem step), and a related feature, feature 1', not requesting help on well-known skills.
2. Long pauses after receiving bug messages (error messages given when the student's behavior indicates a known misconception) which may indicate self-explanation (cf. Chi, Bassok, Lewis, Reimann, & Glaser, 1989) of the bug message, and its inverse, 2', short pauses after receiving bug messages (indicating a failure to self-explain)

3. Long pauses after reading on-demand help messages (potentially indicating deeper knowledge or self-explanation), and an inverse feature, 3', short pauses after reading the on-demand help message
4. Long pauses after reading an on-demand help message and getting the current action right (cf. Shih, Koedinger, & Scheines, 2008), and an inverse feature, 4', short pauses after reading an on-demand hint message and getting the current action right. Features 4 and 4' are sub-sets of features 3 and 3'.
5. Long pauses on skills that the student probably knows (may indicate continuing to self-explain even after proceduralization), and an inverse feature, 5', short pauses on skills assessed as known
6. Off-task behavior (Baker, 2007), where the student is engaged in behavior which does not involve the system or a learning task, and a related feature, 6', long pauses that are not off-task (may indicate self-explanation, or asking teacher for help – cf. Schofield, 1995). Off-task behavior is assessed using an automated detector (Baker, 2007).
7. Gaming the system (Baker, Corbett, Roll, & Koedinger, 2008), attempting to succeed at problem steps without learning the material (by clicking through help messages quickly until receiving the answer, or systematic guessing), and a related feature, 7', fast actions that do not involve gaming (which may indicate a very well-known skill). These features are computed using an automated detector of gaming the system (Baker, Corbett, Roll, & Koedinger, 2008).
8. The student's average probability of contextual slip/carelessness on errors, making an error when the student is assessed to know the relevant skill (known to predict post-test problem-solving performance – Baker et al., 2010). This feature is computed using an

automated detector (Baker et al., 2010). Also, a related feature, 8', the certainty of contextual slip, the average contextual slip computed only for values of contextual slip over 0.5; this represents how certain the model is when it indicates that a student has slipped.

9. The student's average learning per learning opportunity using the moment-by-moment learning model, which estimates the probability that the student learned a relevant skill at each step in problem solving. Also, a related feature, 9', the degree to which there are spikes in learning, defined as the ratio between the maximum moment-by-moment learning and the average moment-by-moment learning.

Many of these features involve a continuous variable, such as the time taken between actions or the probability of knowing a skill. In general, our detectors do not hinge on a student's average value for the feature (e.g., average time between actions), but instead hinge on the proportion of actions that meet a constraint (e.g., the proportion of actions with a short pause, or the proportion of actions with a long pause). For each such feature, we empirically determined a cut-off value that indicates whether the student behavior occurred or not (e.g. a long pause or low probability), rather than averaging the actual values (times or probabilities), in order to avoid having a small proportion of extreme behaviors of interest be overwhelmed by noise in the rest of the student's data.

Once feature engineering had been completed, a three-step process was conducted to develop model of transfers and PFL: selecting features, optimizing feature cut-offs, and combining the features into a unified prediction model. In order to select a set of features, we fit a one-parameter linear regression model predicting transfer from each feature (or related feature), using correlation as the measure of each feature's goodness. In order to increase the probability of a generalizable model, we assessed each model's correlation using student-level leave-out-one-cross-validation (LOOCV). In this approach, a model is repeatedly fit for every student except one, and then goodness-of-fit is tested on the left-out student. Every student is excluded from the training set and used as the test set exactly once. In this situation, each model fit can have either a positive or negative coefficient, therefore the sign of a cross-validated correlation does not imply the direction of a relationship but instead implies its consistency. A positive cross-validated correlation implies that the models generalize across the data, while a negative cross-validated correlation implies that the models fail to generalize across the data (and the relationship actually flips direction for a substantial number of students). Using cross-validation in this fashion is considered a valid alternative to statistical significance testing (cf. Raftery, 1995), which explicitly examines the goodness of the models on new data, rather than investigating how well the model fits the data it is trained on (Efron & Gong, 1983).

Transfer Detector

Only features with positive cross-validated correlation to the transfer or PFL test were considered for inclusion in the full model.

For the transfer detector, nine features met this criterion: 1 (Help Avoidance), with a cut-off of 70% probability for "poorly known"; 2 (Long Pauses after a Bug Message), with a cut-off of 7 seconds for "long"; 2'(Short Pauses after a Bug Message), with a cut-off of 1.5 seconds for "long"; 3 (Long Pauses after a Hint), with a cut-off of 8 seconds for "long"; 4 (Long Pauses after a Hint and Correct Answer), with a cut-off of 12 seconds for "long"; 6 (Off-Task Behavior); 7 (Gaming the System); 7' (Fast Non-Gaming Actions), with a cut-off of 2 seconds for "fast"; and 9' (Spikiness in Moment-by-Moment Learning).

Seven out of nine of these features depend on a threshold parameter, N; adjusting a feature's parameter can result in a very different model. For each of these features, we used brute-force grid search to find an optimal cut-off level for each of the above-mentioned features (in grid search, values are tried for every step at the same interval – for instance 0.5 seconds, 1 second, 1.5 seconds, 2 seconds, etc.). Optimality was defined in terms of the ability to predict the dependent variable, performance on the transfer test. Variables involving probabilities were searched at a grid size of 0.05; variables involving time were searched at a grid size of 0.5 seconds.

The cross-validated correlations for single-feature regression models are shown in Table 1.

(Place Table 1 approximately here)

These nine features were considered as potential candidates for a unified model (other features, which individually had cross-validated correlations below zero, were eliminated from consideration, as a control on over-fitting). To find a unified model combining multiple parameters, Forward Selection was conducted (Ramsey & Schafer, 1997). In Forward Selection, the best single-parameter model is chosen, and then the parameter that most improves the model is repeatedly added until no more parameters can be added which improve the model. The goodness metric used was the LOOCV correlation between the predictions and each student's performance on the transfer test.

The resultant model was:

Transfer = -1.5613 * HelpAvoidance(1) + 0.2968 * FastNotGaming(7') + 0.8272

The feature most strongly associated with transfer, both by itself and as a component of a unified model, was avoiding help, which was negatively associated with transfer (cross-validated r=0.376). One potential interpretation is that help avoidance directly caused lower learning (cf. Aleven et al., 2006), perhaps causing the students to have less conceptual learning, as the tutor hints are fairly conceptual in nature. This lack of conceptual understanding may in turn have made these students less able to transfer their knowledge. The other individual feature incorporated into the model was fast non-gaming actions. These actions were significantly positively associated with transfer. Fast non-gaming actions may indicate a degree of fluency with the relevant skills that facilitates reasoning with them, as hypothesized by Haverty, Koedinger, Klahr, and Alibali (2000), leading to better transfer.

The cross-validated correlation of the model to the transfer test was 0.396, as shown in Table 2.

(Place Table 2 approximately here)

PFL Detector

The same set of 18 features and model development process described in the previous section was used to develop a model of students' preparation for future learning. In the case of PFL, five features showed positive cross-validated correlations between the individual feature and the students' performance on the PFL test: 1 (Help Avoidance), with a cut-off of 85% probability for "poorly known"; 3 (Long Pauses after a Hint), with a cut-off of 8 seconds for "long"; 3' (Short Pauses after a Hint), with a cut-off of 1 second for "short"; 4 (Long Pauses after a Hint and Correct Answer), with a cut-off of 8 seconds for "long"; 4' (Short Pauses after a Hint and Correct Answer), with a cut-off of 20 seconds for "short"; 6 (Off-Task Behavior); 6' (Long

Pauses that are not Off-Task), with a cut-off of 4 seconds for “long”; 7 (Gaming the System), 7’ (Fast Non-Gaming Actions), with a cut-off of 4 seconds for “fast”; 9 (Average Moment-by-Moment Learning); and 9’ (Spikiness in Moment-by-Moment Learning).

Single-feature regression models fit on the whole data set and their associated cross-validated correlations are shown in Table 3 (only features with cross-validated correlation over 0 are shown).

(Place Table 3 approximately here)

These eleven features were considered as potential candidates for a unified model. To find a unified model combining multiple parameters, Forward Selection was conducted, as with the transfer model.

The resultant models was:

PFL = 0.0127 * Spikiness(9) - 0.5499 * HelpAvoidance(1) - 5.3898 * LongPauseAfterHint(4) + 0.8773

The feature most strongly associated with PFL was long pauses after reading hint messages and getting the next action correct, which was somewhat unexpectedly negatively associated with PFL (cross-validated r=0.410). As with transfer, help avoidance was also negatively associated with PFL (cross-validated r=0.329), and entered into the final model. Finally, the spikiness of the student’s learning is positively associated with PFL, and enters into the final model, achieving a cross-validated r of 0.233. This finding suggests that PFL is higher if a student’s learning more frequently occurs in relatively sudden “aha” moments, as compared to occurring more gradually, deeper learning is occurring.

As shown in Table 2, the overall cross-validated correlation of the model to the PFL test was 0.454.

Transfer and PFL

Given the existence of models that can predict PFL and transfer to a reasonable degree, one question is to what degree these two models are capturing the same construct. The two constructs have a fairly substantial correlation of 0.520. However, it is worth studying whether the two forms of robust learning are characterized by the same behaviors during learning.

The results of these two models seem to suggest substantial overlap. First, several of the same data features were found to be associated with both transfer and PFL under cross-validation: 1, 3, 4, 5, 7, 7', and 9'. In fact, only 2 features predicted transfer but failed to predict PFL, and only 4 features predicted PFL but failed to predict transfer.

In addition, each model was successful at predicting the other construct. When used to predict PFL, the optimized-feature transfer detector achieves a correlation of 0.425, almost as good as the optimized model trained to predict PFL. Correspondingly, when used to predict transfer, the optimized-feature PFL detector achieves a correlation of 0.395, almost identical to the detector trained just to predict Transfer.

Studying the Goodness of Transfer and PFL Detectors for High School Data

After developing these detectors, our next goal was to understand how well these detectors transfer between different populations of students. To this end, data was analyzed for a sample of high school students working with the same Genetics Cognitive Tutor module, to examine whether the robust learning models transfer between two populations who vary in age and prior preparation.

Data Set

As in the original study, the data used in the second study came from the Genetics Cognitive Tutor Three-Factor Cross module. Fifty-six high school students who were enrolled in high school biology courses used the tutor. The students were recruited to participate in the study for pay through several methods, including advertisements in a regional newspaper and recruitment handouts distributed at two urban high schools.

The study had the same design as the college-level study. In specific, it consisted of two 2-hour sessions, followed by a shorter session one week later, all conducted in computer clusters at Carnegie Mellon University. The students engaged in Cognitive Tutor-supported activities for one hour in each of two sessions. As in the original study, students completed a transfer test and preparation for future learning test after using the tutor, as well as completing a pre-test and post-test of the exact skills taught in the tutor. All tests were identical to the ones used in the previous study.

The 56 students completed a total of 21,498 problem solving attempts across a total of 9204 problem steps in the tutor. The number of problem-solving attempts per student was not significantly different between the college and high school populations, $t(126)=0.847$, two-tailed $p=0.40$. Like the college students, the high school students demonstrated successful learning in this tutor, with an average pre-test performance of 0.16 (SD=0.09) and an average post-test performance of 0.56 (SD=0.28), a statistically significant difference, $t(55)=11.443$, two-tailed $p<0.001$. Students' average transfer test performance was 0.53 (SD=0.22) and average PFL performance was 0.66 (SD=0.28).

Transferring Robust Learning Detectors From College Students to High School Students

To check the generalizability of the transfer and PFL detectors, we tested the predictive power of each detector, taking the detectors developed and optimized using the college data and applying them without modification to the high school data set.

The college detector of transfer achieved a correlation of 0.426 to the transfer test scores within the high school data set. It is worth noting that this correlation was higher than the correlation (0.396) in the college data set, despite the model being transferred to a new population. One possible explanation is that there is a closer link between in-tutor performance and transfer test performance in the high school population than the college population, potentially because students were closer to reaching the performance ceiling in the original college population.

By contrast, the college detector of PFL achieved a correlation of 0.228 to the PFL test scores within the high school data set, a value that represents substantial degradation compared to the data set for which these models was originally developed (where the value was 0.454). At the same time, this model remains marginally statistically significantly higher than zero, two-tailed p=0.09.

Building New Robust Learning Detectors For High School Students

In order to fully understand the degree of degradation between the college and high school populations, we can build new detectors for the high school population. Seeing how well these detectors perform can give us an upper limit for how well this type of detector can perform in this data set. It also may be interesting to study which data features are important predictors within the high school population, to see how these features differ from those used in the college population, at a qualitative level.

A new detector of transfer trained on the data from the high school population using optimized features achieves a cross-validated correlation of 0.528. This number is moderately higher than the goodness of the detector trained on the college population and then applied to this data set, which was 0.426. It is also higher than the performance of the goodness of the detector trained on the college population on its original data set, which was 0.396, again indicating that student behavior is more closely linked to performance on the transfer test in the high school population than in the college population.

By contrast, a new detector of PFL trained on the data from the high school population using optimized features achieves an unimpressive cross-validated correlation of 0.181. This number is actually lower than the goodness of the detector trained on the college population and then applied to this data set, which was 0.228. It is also substantially lower than the performance of the goodness of the detector trained on the college population on its original data set, which was 0.454. This result indicates that the behaviors associated with PFL in this new population are not captured well by the feature set originally developed within the college population.

Features Associated with Robust Learning in High School Data Set: Transfer

Within the high school data set, thirteen individual features were found to have positive cross-validated correlation to the transfer test scores. The single feature linear regression model for each feature is given in Table 4.

(Place Table 4 approximately here)

There was substantial overlap between the features that had positive cross-validated correlations in the college and high school populations. Only one of the features that had a positive cross-validated correlation for the college population failed to have a positive cross-validated correlation for the high school population, short pauses after bug messages (Feature 2). Of the remaining features, all but two pointed in the same direction in both data sets (pointing in the same direction means that the model coefficient was either negative in both data sets or positive in both data sets). The two which changed direction were the spikiness of moment-by-moment learning (negative in the college data set and positive in the high school data set), and off-task behavior (negative in the college data set and positive in the high school data set). It is worth noting that off-task behavior had the weakest relationship that still had a positive cross-validated correlation, in both data sets (0.024 and 0.051). Hence, the primary noteworthy difference is the relationship for spikiness.

That said, it is worth noting that many of the features changed semantics substantially during parameter optimization. Only one feature retained similar semantics between the two data sets, help avoidance (Feature 1), which had an optimized cut-off of 70% in the college data set, but an optimized cut-off of 50% in the high school data set, a relatively minor change. In terms of features that changed semantics, feature 3, long pauses after reading help messages changed from a cut-off of 8 seconds in the college data set to 1 second in the high school data set, a substantially different feature. Similarly, feature 4, long pauses after reading help messages and then obtaining a correct answer, changed from 12 seconds in the college data set to 1 second in the high school data set. Feature 7’, fast non-gaming actions, shifted in the other direction, from 2 seconds to 20 seconds.

Five additional features were also significant in the high school model: 3’ (Short Pauses after a Hint), with a cut-off of 17 seconds for “short”; 4’ (Short Pauses after a Hint and Correct Answer), with a cut-off of 17 seconds for “short”; 8 (Average Contextual Slip); 8’ (Certainty of Contextual Slip); 9 (Average Moment-by-Moment Learning).

A model was fit using Forward Selection, as in the college data set. The best model of transfer for the high school data set, using the optimal feature cut-offs, and fitting to all data, was as follows:

Transfer = -0.793 * Gaming(7) + 1.518 * Off-task behavior(6) -34.429 * LongPauseAfterBug(2) + 0.7587

Features Associated with Robust Learning in High School Data Set: PFL

A range of variables were found to be have cross-validated correlations over 0 to the PFL test within the high school population, shown in Table 5. There was considerable overlap between the college and high school populations for these features. 7 of the 11 features used in the college detector of PFL were also used in the high school detector of PFL (3, 3’, 4, 7, 7’, 9, 9’), with all pointing in the same direction in the two data sets except for 3’, which switched direction.

(Place Table 5 approximately here)

However, none of these features had particularly impressive correlations taken individually, with the highest cross-validated correlation for the high school data set having a value of 0.137. This feature was feature 9', the spikiness of the moment-by-moment learning model. Two other features had cross-validated correlations of 0.1 or higher: the certainty of slip, and gaming the system. Spikiness and gaming were also found in the college PFL model, where the relationships pointed in the same direction as in the high school data set.

A model of PFL was fit using Forward Selection, as in the college data set. The best model of PFL for the high school data set, using the optimal feature cut-offs, and fitting to all data, was as follows:

PFL = 0.0288 * Spikiness (9') – 1.1901 * LongPauseAfterHint (3) – 27.343 * LongPauseAfterBug (2) + 0.6214

Conclusions

In this paper, we have studied the degree to which automated detectors of transfer and preparation for future learning transfer to a new cohort of students, using the same tutor lesson. These findings establish that it is not just possible to identify whether a student has achieved robust learning; it is also possible to successfully apply these models on a different population than the initial population these detectors were developed for, establishing that there is some degree of generality in the constructs that these detectors tap.

The detector of transfer generalized from the college population to the high school population with limited evidence of degradation; in fact, the detector functioned better within the new population than in the original population, though not quite as well as a new detector trained specifically for the new population.

The detector of PFL, on the other hand, saw relatively greater evidence of degradation between the college and high school population, achieving a correlation only about half as high within the high school population as had been achieved within the college population. However, it may just be that PFL was relatively difficult to detect within the high school population, as a detector trained specifically for the new population also functioned relatively poorly.

Between the high school and college populations, many of the same features were predictive of transfer and PFL. There was substantial overlap in both cases, with 7 of 9 features that had cross-validated correlation over 0 in the college data set achieving a cross-validated correlation over 0 and a coefficient pointing in the same direction as in the college model, when transferred to the high school data set. 6 of 11 features achieved this same standard when the college model of PFL was transferred to the high school data set, a lower degree of overlap but still an indication of considerable similarity between the construct in the two data sets.

Four features were predictive (and pointed in the same direction) in every model: 3, 4, 7, and 7'. Feature 3, long pauses after reading hint messages, and Feature 4, long pauses after reading hint messages and providing a correct answer, were negatively correlated with robust learning for each construct and data set. This does not necessarily mean that these pauses (interpreted as implying self-explanation – cf. Shih, Koedinger, & Scheines, 2008) actually hurt learning, but may instead indicate a general selection bias where the students who seek help are generally less knowledgeable (cf. Aleven et al., 2006). These results build on past findings regarding relationships between students' strategies for using help and their learning outcomes (cf. Aleven et al., 2006). We recommend that future research on help-seeking and learning consider measures of transfer and preparation for future learning to a greater degree.

Feature 7, gaming the system, was also negatively correlated with robust learning for each construct and data set, albeit with relatively low correlations. This finding accords with previous results suggesting that gaming the system is particularly pernicious for learning (cf. Cocea, Hershkovitz, & Baker, 2009).

However, fast non-gaming actions were positively correlated with robust learning for each construct and data set, with generally strong correlations. These actions appear to indicate robust learning that leads to both transfer and PFL. Given that fast correct actions are also associated with retention (cf. Pavlik & Anderson, 2008), it appears that rapid correct performance indicates learning that is robust in multiple fashions.

Many other features were associated with robust learning for a single construct. Help avoidance was associated with transfer with a strong negative correlation in both populations. Previous analysis has also found negative correlations between help avoidance and learning – e.g. students who make errors when they should have sought help perform more poorly on tests of standard problem-solving (Aleven et al., 2006). Help in the Genetics Cognitive Tutor is fairly conceptual in nature; that is, it relates the steps in the problem-solving procedure to the properties of the underlying genetic processes. Our findings suggest that this type of help is associated not just with learning to solve the types of problems in the tutor, but leads to robust learning as well. Prior work studying the learning impact of teaching students when to seek help has not had significant effects on problem-solving post-tests (Roll, Aleven, McLaren, & Koedinger, 2011); it would be worth studying whether this type of meta-cognitive instruction impacts performance on measures of robust learning, even if it does not impact performance on problem-solving post-tests. An alternate explanation for the negative relationship between help avoidance and robust learning in our study – in line with the results in (Roll et al., 2011) – is that some students are not prepared to learn from the types of help in the tutor, leading them to both avoid help and demonstrate less robust learning. In general, further attention to why students avoid help, and how students use help successfully and unsuccessfully (cf. Aleven, Stahl, Schworm, Fischer, & Wallace, 2003) may help us understand this finding better.

Features of the moment-by-moment learning model were associated with PFL in both data sets. In particular, spikiness as measured by the moment-by-moment learning model was positively associated with PFL in both data sets. In other recent work, it has been suggested that further distillations of the moment-by-moment graph, in particular through explicitly considering the visual form of the graph, can be even more predictive of student preparation for future learning (Baker et al., in press).

In general, this paper suggests that models of robust learning can be transferred to new populations. As such, these models can be used with relative confidence for new groups of students, to drive interventions. By doing so, we can move towards the vision of learning systems which can adapt effectively to individual differences not just in what students know, but in how robust their learning is.

Another valuable area of future work will be to determine how general the phenomena seen here are for new content: new lessons within the Genetics Tutor, Cognitive Tutors on other topics, and additional learning systems. The models presented here are time-consuming in nature to develop; to the extent that general models can be developed, their potential usefulness will be substantially increased.

Acknowledgements

We would like to thank Robert Siegler, Ken Koedinger, Kurt VanLehn, Ben MacLaren, Lisa Rossi, and Belinda Yew for helpful comments and suggestions. This research was supported via grant "Empirical Research: Emerging Research: Robust and Efficient Learning: Modeling and Remediating Students' Domain Knowledge", National Science Foundation award number #DRL-0910188, via grant "Promoting Robust Understanding of Genetics with a Cognitive Tutor that Integrates Conceptual Learning with Problem Solving", Institute of Education Sciences award number #R305A090549, and by the Pittsburgh Science of Learning Center, NSF award number #SBE-0836012. Portions of the work published here were previously presented in conference papers (Baker, Gowda, & Corbett, 2011a, 2011b), listed below.

References

Aleven, V., & Koedinger, K.R. (2002). An effective metacognitive strategy: learning by doing and explaining with a computer-based Cognitive Tutor. *Cognitive Science*, 26, 147-179.

Aleven, V., McLaren, B., Roll, I., & Koedinger, K. (2006). Toward meta-cognitive tutoring: A model of help seeking with a Cognitive Tutor. *International Journal of Artificial Intelligence and Education*, 16, 101-128.

Aleven, V., Stahl, E., Schworm, S., Fischer, F., & Wallace, R.M. (2003). Help seeking and help design in interactive learning environments. *Review of Educational Research*, 73(2), 277-320.

Atkinson, R. K. (2002). Optimizing learning from examples using animated pedagogical agents. *Journal of Educational Psychology*, 94, 416-427.

Bahrick, H.P., Bahrick, L.E., Bahrick, A.S., & Bahrick, P.E. (1993). Maintenance of Foreign Language Vocabulary and the Spacing Effect. *Psychological Science*, 4(5), 316-321.

Baker, R.S.J.d. (2007). Modeling and Understanding Students' Off-Task Behavior in Intelligent Tutoring Systems. *Proceedings of ACM CHI 2007: Computer-Human Interaction*, 1059-1068.

Baker, R.S.J.D., Corbett, A.T, & Aleven, V. (2008). More Accurate Student Modeling Through Contextual Estimation of Slip and Guess Probabilities in Bayesian Knowledge Tracing. *Proceedings of the 9th International Conference on Intelligent Tutoring Systems*, 406-415.

Baker, R.S.J.d., Corbett, A.T., Gowda, S.M., Wagner, A.Z., MacLaren, B.M., Kauffman, L.R., Mitchell, A.P., & Giguere, S. (2010). Contextual Slip and Prediction of Student Performance After Use of an Intelligent Tutor. *Proceedings of the 18th Annual Conference on User Modeling, Adaptation, and Personalization*, 52-63.

Baker, R.S.J.d., Corbett, A.T., Roll, I., & Koedinger, K.R. (2008). Developing a Generalizable Detector of When Students Game the System. *User Modeling and User-Adapted Interaction*, 18(3), 287-314.

Baker, R.S.J.d., Gowda, S., Corbett, A.T. (2011a) Towards predicting future transfer of learning. *Proceedings of 15th International Conference on Artificial Intelligence in Education*, 23-30.

Baker, R.S.J.d., Gowda, S.M., Corbett, A.T. (2011b) Automatically Detecting a Student's Preparation for Future Learning: Help Use is Key. *Proceedings of the 4th International Conference on Educational Data Mining*, 179-188.

Baker, R.S.J.d., Hershkovitz, A., Rossi, L.M., Goldstein, A.B., Gowda, S.M. (in press) Predicting Robust Learning with the Visual Form of the Moment-by-Moment Learning Curve. To appear in the *Journal of the Learning Sciences.*

Baker, R.S.J.d., Walonoski, J., Heffernan, N., Roll, I., Corbett, A., & Koedinger, K. (2008). Why Students Engage in “Gaming the System” Behavior in Interactive Learning Environments. *Journal of Interactive Learning Research*, *19*, 185-224.

Bransford, J.D., Schwartz, D.L. (1999) Rethinking transfer: a simple proposal with multiple implications. *Review of Research in Education, 24*, 61-100.

Butcher, K.R. (2010). How Diagram Interaction Supports Learning: Evidence from Think Alouds During Intelligent Tutoring. *LNCS*, 6170, 295-297.

Chi, M.T.H., Bassok, M., Lewis, M.W., Reimann, P., & Glaser, R. (1989). Self-explanations: how students study and use examples in learning to solve problems. *Cognitive Science*, 13, 145-182.

Chi, M.T.H, Feltovich, P.J. & Glaser, R. (1981). Categorization and representation of physics problems by experts and novices. *Cognitive Science*, 18, 439-477.

Chi, M., & VanLehn, K. (2007). Domain-Specific and Domain-Independent Interactive Behaviors in Andes. In R. Luckin, K. R. Koedinger & J. Greer (Eds.) *Proceedings of the International Conference on Artificial Intelligence in Education*. pp. 548-550. Amsterdam, Netherlands: IOS Press.

Chin, D.B., Dohmen, I.M., Cheng, B.H., Oppezzo, M.A., Chase, C.C., & Schwartz, D. L. (2010). Preparing Students for Future Learning with Teachable Agents. *Educational Technology Research and Development*, 58(6), 649-669.

Cocea, M., Hershkovitz, A., Baker, R.S.J.d. (2009) The Impact of Off-task and Gaming Behaviors on Learning: Immediate or Aggregate? *Proceedings of the 14th International Conference on Artificial Intelligence in Education*, 507-514.

Conati, C., Gertner, A., & VanLehn, K. (2002). Using Bayesian networks to manage uncertainly in student modeling. *User Modeling & User-Adapted Interaction*, 12(4), 371-417.

Corbett, A. (2001). Cognitive computer tutors: Solving the two-sigma problem. In M. Bauer, P. Gmytrasiewicz, & J. Vassileva (Eds.), UM2001, *User Modeling: Proceedings of the Eighth International Conference* (pp. 137–147). Berlin: Springer.

Corbett, A.T., & Anderson, J.R. (1995). Knowledge Tracing: Modeling the Acquisition of Procedural Knowledge. *User Modeling and User-Adapted Interaction*, 4, 253-278.

Corbett A., & Bhatnagar A. (1997). Student Modeling in the ACT Programming Tutor: Adjusting Procedural Learning Model with Declarative Knowledge. User Modeling: *Proceedings of the 6th International Conference*, 243-254.

Corbett A.T., Maclaren, B., Kauffman, L., Wagner, A., & Jones, E. (2010). A Cognitive Tutor for genetics problem solving: Learning gains and student modeling. *Journal of Educational Computing Research*, 42, 219-239.

Corbett, A., MacLaren, B., Wagner, A., Kauffman, L., Mitchell, A., Baker, R. and Gowda, S. (2011). Preparing students for effective explaining of worked examples in the Genetics Cognitive Tutor. In *Proceedings of the Thirty-third Annual Meeting of the Cognitive Science Society*.

Corbett, A. T., & Trask, H. (2000). Instructional interventions in computer-based tutoring: differential impact on learning time and accuracy. In *Proceedings of the ACM CHI ’2000 Conerence on Human Factors in Computing Systems* (pp. 97–104).

Craig, S., VanLehn, K. & Chi, M. (2008). Promoting learning by observing deep-level reasoning questions on quantitative physics problem solving with Andes. In K. McFerrin et al. (Eds.), *Proceedings of Society for Information Technology and Teacher Education International Conference 2008* (pp. 1065-1068). Chesapeake, VA: AACE.

Desmarais, M. C., Meshkinfam, P., & Gagnon, M. (2006). Learned Student Models with Item to Item Knowledge Structures. *User Modeling and User-Adapted Interaction*, 16(5), 403-434.

Duncker, K. (1945). On Problem Solving. *Psychological Monographs*, 58, 270.

Efron, B. & Gong, G. (1983). A leisurely look at the bootstrap, the jackknife, and cross-validation. *American Statistician*, 37, 36-48.

Fong, G. T., & Nisbett, R. E. (1991). Immediate and delayed transfer of training effects in statistical reasoning. *Journal of Experimental Psychology: General*, 120, 34-45.

Gick, M.L., & Holyoak, K.J. (1987). The cognitive basis of knowledge transfer. Cormier, S.M. (Ed); Hagman, J.D. (Ed), *Transfer of learning: Contemporary research and applications. The educational technology series.* (pp. 9-46). San Diego, CA, US: Academic Press.

Gong, Y., Beck, J. E., & Heffernan, N. T. (2010). Comparing Knowledge Tracing and Performance Factor Analysis by Using Multiple Model Fitting Procedures. *Proceedings of the 10th International Conference on Intelligent Tutoring Systems*, 35-44.

Graesser, A. C., Lu, S., Jackson, G. T., Mitchell, H.H.,Ventura, M., Olney, A., & Louwerse, M.M. (2004). AutoTutor: A tutor with dialogue in natural language. *Behavioral Reseach Methods, Instruments and Computers*, *36*, 180-193.

Hausmann, R. & VanLehn, K. (2007). Explaining Self-Explaining: A Contrast Between Content and Generation. *Proceedings of the 13th International Conference on Artificial Intelligence in Education*, pp. 417-424.

Haverty, L.A., Koedinger, K.R., Klahr, D., Alibali, M.W. (2000) Solving inductive reasoning problems in mathematics: not-so-trivial pursuit. *Cognitive Science*, *24* (2), 249-298.

Jastrzembski, T. S., Gluck, K. A., & Gunzelmann, G. (2006). Knowledge tracing and prediction of future trainee performance. In *Proceedings of the 2006 Interservice/Industry Training, Simulation, and Education Conference* (pp. 1498-1508).

Katz, S., Connelly, J., & Wilson, C. (2007). Out of the lab and into the classroom: An evaluation of reflective dialogue in ANDES. In *Proceedings of the 2007 conference on Artificial Intelligence in Education*, 425–432. Amsterdam, The Netherlands: IOS Press

Koedinger, K. R., & Corbett, A. T. (2006) Cognitive tutors: Technology bringing learning sciences to the classroom. In R. K. Sawyer (Ed.), *The Cambridge handbook of the learning sciences*, pp. 61-77. New York, NY: Cambridge University Press.

Koedinger, K.R., Corbett, A.T., & Perfetti, C. (2012). The Knowledge-Learning-Instruction (KLI) Framework: Toward Bridging the Science-Practice Chasm to Enhance Robust Student Learning. *Cognitive Science*, *36*, 757-798.

Li, N., Cohen, W.W., & Koedinger, K.R. (2010). A computational model of accelerated future learning through feature recognition. In *Proceedings of the International Conference on Intelligent Tutoring Systems* (pp. 368-370).

Martin, J., & VanLehn, K. (1995). Student Assessment Using Bayesian Nets. *International Journal of Human-Computer Studies*, 42, 575-591.

Mathan, S.A., & Koedinger, K.R. (2005). Fostering the Intelligent Novice: Learning from errors with metacognitive tutoring. *Educational Psychologist*. 40(4), 257-265.

McCalla, G. (1992). The Search for Adaptability, Flexibility and Individualization: Approaches to Curriculum in ITS. In: M. Jones and P. Winne (eds.) *Adaptive Learning Environments: Foundations and Frontiers.* Springer-Verlag, Berlin, pp. 91-122.

McLaren, B.M., Lim, S., & Koedinger, K.R. (2008). When and how often should worked examples be given to students? New results and a summary of the current state of research. *Proc. of the 30th Annual Conf. of the Cognitive Science Society,* p. 2176-2181.

Pardos, Z. A., Gowda, S. M., Baker, R.S.J.d., Heffernan, N. T. (2011) Ensembling Predictions of Student Post-Test Scores for an Intelligent Tutoring System. *Proceedings of the 4th International Conference on Educational Data Mining*, 189-198

Pavlik, P.I., & Anderson, J.R. (2008). Using a Model to Compute the Optimal Schedule of Practice. *Journal of Experimental Psychology: Applied*, 14(2), 101-117.

Pavlik, P.I., Cen, H., & Koedinger, J.R. (2009). Performance Factors Analysis – A New Alternative to Knowledge Tracing. *Proceedings of the 14th International Conference on Artificial Intelligence in Education,* 531-540.

Pavlik, P.I., Cen, H., Wu, L., & Koedinger, K.R. (2008). Using Item-type Performance Covariance to Improve the Skill Model of an Existing Tutor. In: Baker, R.S., Beck, J.E. (Eds.) *Proceedings of the 1st International Conference on Educational Data Mining*. Montreal, Canada, p. 77-86.

Raftery, A.E. (1995). Bayesian model selection in social research (with discussion). *Sociological Methodology*, 25, 111-195.

Ramsey, R.L., & Schafer, D.W. (1997). *The Statistical Sleuth*. Belmont, CA: Wadsworth Publishing.

Rittle-Johnson, B., & Siegler, R.S. (1998). The relation between conceptual and procedural knowledge in learning mathematics: A review. In C. Donlan (Ed.), *The development of mathematical skill* (pp. 75-110). Hove, UK: Psychology Press.

Roll, I., Aleven, V., McLaren, B. M., & Koedinger, K. R. (2007). Designing for metacognition - applying cognitive tutor principles to the tutoring of help seeking. *Metacognition and Learning*, 2(2), 125-140.

Roll, I., Aleven, V., McLaren, B.M., & Koedinger, K.R. (2011). Improving students' help-seeking skills using metacognitive feedback in an intelligent tutoring system. *Learning and Instruction*, 21, 267-280.

Salden, R.J.C.M., Koedinger, K.R., Renkl, A., Aleven, V., & McLaren, B.M. (2010). Accounting for Beneficial Effects of Worked Examples in Tutored Problem Solving. *Educational Psychology Review*, 22, 379-392.

San Pedro, M.O.C.Z., Baker, R.S.J.d., & Rodrigo, M.M.T. (2011). Detecting Carelessness through Contextual Estimation of Slip Probabilities among Students Using an Intelligent

Tutor for Mathematics. *Proceedings of 15th International Conference on Artificial Intelligence in Education*, 304-311.

Schmidt, R.A., & Bjork, R.A. (1992). New conceptualizations of practice: common principles in three paradigms suggest new concepts for training. *Psychological Science*, 3(4), 207-217.

Schofield, J.W. (1995). *Computers and Classroom Culture*. Cambridge, UK: Cambridge University Press.

Schwartz, D.L. & Martin, T. (2004). Inventing to prepare for future learning: The hidden efficiency of encouraging original student production in statistics instruction. *Cognition and Instruction*, 22, 129-184.

Schwonke, R., Renkl, A., Krieg, C., Wittwer, J., Aleven, V., & Salden, R. (2009). The worked-example effect: Not an artefact of lousy control conditions. *Computer in Human Behavior,* 25, 258–266.

Shih, B., Koedinger, K.R., & Scheines, R. (2008). A response time model for bottom-out hints as worked examples. *Proceedings of 1st International Conference on Educational Data Mining*, 117-126.

Shute, V.J. (1995). SMART: Student modeling approach for responsive tutoring. *User Modeling and User-Adapted Interaction*, 5(1), 1-44.

Singley, M.K., & Anderson, J.R. (1989). *The Transfer of Cognitive Skill*. Cambridge, MA: Harvard University Press.

Tan, J., & Biswas, G. (2006). The Role of Feedback in Preparation for Future Learning: A Case Study in Learning by Teaching Environments. *Proceedings of the International Conference on Intelligent Tutoring Systems*, 370-381.

VanLehn, K. (2006). The Behavior of Tutoring Systems. *International Journal of Artificial Intelligence in Education*, 16 (3), 227-265.

VanLehn, K., Jones, R., Chi, M.T.H. (1992) A Model of the Self-Explanation Effect. *Journal of the Learning Sciences, 2* (1), 1-59.

Wang, Y. & Heffernan, N. (2011) The "Assistance" Model: Leveraging How Many Hints and Attempts a Student Needs. *Proceedings of the 24th International FLAIRS Conference*.

Table 1. Goodness of single-feature linear regression models for predicting transfer in the college data set.

Feature	Transfer =	Cross-validated r
1) Help avoidance	-1.735 * F1 + 0.912	0.376
9') Spikiness of moment-by-moment learning	-9.758 * F9 + 0.951	0.346
4) Long pauses after reading hint messages and then getting the next action right	-6.510 * F4 + 0.893	0.204
3) Long pauses after reading hint messages	-4.075 * F3 + 0.902	0.199
7') Fast actions that do not involve gaming	0.484 * F7' + 0.726	0.188
2) Long pauses after receiving bug messages	-13.497 * F2 + 0.880	0.130
7) Gaming the system	-0.2058 * F7 + 0.903	0.076
2') Short pauses after receiving bug messages	-4.291 * F2' + 0.876	0.037
5) Off-task behavior	-1.037 * F5 + 0.899	0.024

Table 2. Cross-validated correlations between models and tests.

Construct	Data Developed With	Data Tested On	Cross-Validated Correlation
Transfer	College	College	0.396
Transfer	College	High School	0.426
Transfer	High School	High School	0.528
PFL	College	College	0.454
PFL	College	High School	0.228
PFL	High School	High School	0.181

Table 3. Goodness of single-feature linear regression models for predicting PFL in the college data set.

Feature	PFL=	Cross-validated r
4) Long pauses after reading hint message(s) and then getting the next action right	-7.67 * F4 + 0.961	0.410
3) Long pauses after reading hint messages	-5.050 * F3 + 0.956	0.376
9) Average moment-by-moment learning	-8.240 * F9 + 0.979	0.345
1) Help avoidance	-1.118 * F1 + 0.952	0.329
9’) Spikiness of moment-by-moment learning	0.022 * F9 + 0.740	0.233
4') Short pauses after reading hint message(s) and then getting the next action right	-1.801 * F4’ + 0.937	0.201
7') Fast actions that do not involve gaming;	0.350 * F7’ + 0.739	0.187
5) Off-task behavior	-1.089 * F5 + 0.944	0.089
5') Long pauses that are not off-task	-0.211 * F5’ + 0.976	0.083
3') Short pauses after reading hint messages	0.173 * F3’ + 0.886	0.034
7) Gaming the system	-0.134 * F7 + 0.93	0.008

Table 4. Goodness of optimized single-feature linear regression models at predicting transfer in high school data set.

Feature	Transfer =	Cross-validated r
7. Gaming the System	-0.9108 * F7 + 0.8482	0.496
9. Average moment-by-moment learning	-16.6448 * F9 + 0.906	0.490
7`. Fast actions that do not involve gaming	0.8805 * F7` + 0.0374	0.437
8. Average Contextual Slip	1.4064 * F8 + 0.0226	0.429
8’. Certainty of Slip	0.8412 * F8 + 0.2947	0.409
3`. Short pauses after reading hint messages	-1.2538 * F3` + 0.6355	0.396
3. Long pauses after reading hint messages	-1.3839 * F3 + 0.6512	0.391
1. Help avoidance	-1.6946 * F1 + 0.7475	0.386
4. Long pauses after reading hint message(s) and then getting the next action right	-1.5936 * F4 + 0.6321	0.367
9’. Spikiness of moment-by-moment learning	0.0598 * F9 + 0.2722	0.362
4`. Short pauses after reading hint message(s) and then getting the next action right	-1.3071 * F4` + 0.61	0.350
2. Long pauses after bug messages	-43.8096 * F2 + 0.5588	0.200
5. Off-task behavior	1.7228 * F5 + 0.4554	0.051

Table 5. Goodness of optimized single-feature linear regression models at predicting PFL in high school data set.

Feature	PFL=	Cross-validated r
9'. Spikiness of moment-by-moment learning	0.045 * F9' + 0.4622	0.137
8'. Certainty of Slip	0.5802 * F8' + 0.4941	0.123
7. Gaming the System	-0.5002 * F7 + 0.8316	0.105
3. Long pauses after reading hint messages	-1.637 * F3 + 0.752	0.097
9. Average moment-by-moment learning	-9.195 * F9 + 0.865	0.092
4. Long pauses after reading hint message(s) and then getting the next action right	-2.3075 * F4 + 0.7452	0.073
2. Long pauses after bug messages	-30.6071 * F2 + 0.6819	0.059
3`. Short pauses after reading hint messages	-0.744 * F3' + 0.7193	0.049
8. Average Contextual Slip	0.7828 * F8 + 0.3744	0.045
7`. Fast actions that do not involve gaming	0.4773 * F7' + 0.3899	0.041

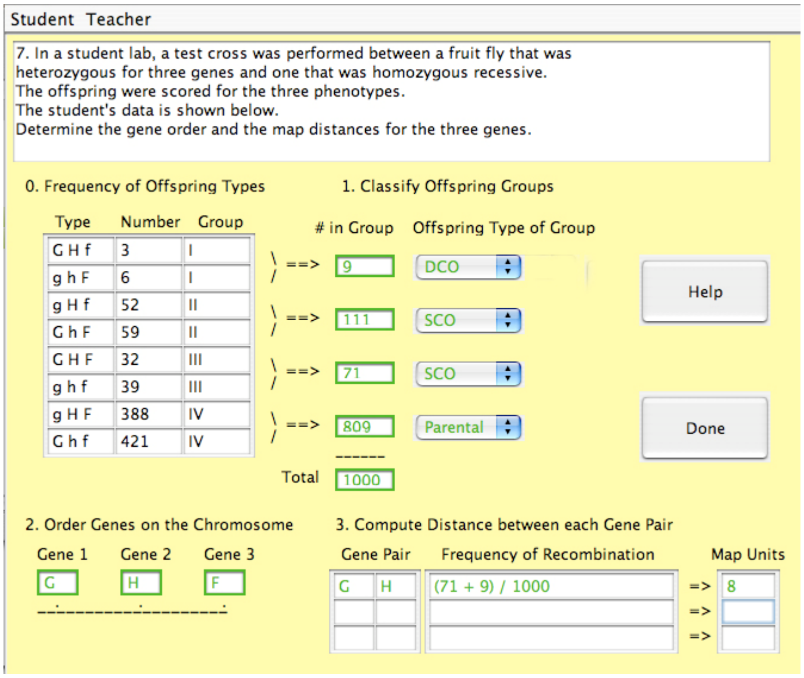

Fig. 1. The Three-Factor Cross lesson of the Genetics Cognitive Tutor.