Using Knowledge Tracing in a Noisy Environment to Measure Student Reading Proficiencies

Joseph E. Beck, *Machine Learning Department, School of Computer Science, Carnegie Mellon University, Pittsburgh, PA 15213. USA*
joseph.beck@gmail.com

June Sison, *Language Technologies Institute, School of Computer Science, Carnegie Mellon University, Pittsburgh, PA 15213. USA*

Abstract. Constructing a student model for language tutors is a challenging task. This paper describes using knowledge tracing to construct a student model of reading proficiency and validates the model. We use speech recognition to assess a student's reading proficiency at a subword level, even though the speech recognizer output is at the level of words and is statistically noisy. Specifically, we estimate the student's knowledge of 80 letter to sound mappings, such as *ch* making the sound /K/ in "chemistry." At a coarse level, the student model did a better job at estimating reading proficiency for 47.2% of the students than did a standardized test designed for the task. Although not quite as strong as the standardized test, our assessment method can provide a report on the student at any time during the year and requires no break from reading to administer. Our model's estimate of the student's knowledge on individual letter to sound mappings is a significant predictor of whether he will ask for help on a particular word. Thus, our student model is able to describe student performance both at a coarse- and at a fine-grain size.

Keywords. Student modeling, evaluation, automated speech recognition

INTRODUCTION

Intelligent Tutoring Systems (ITS) derive much of their power from having a student model (Woolf, 1992) that describes the learner's proficiencies at various aspects of the domain to be learned. For example, the student model can be used to determine what feedback to give (Conati, Gertner, & VanLehn, 2002) or to have the students practice a particular skill until it is mastered (Corbett & Anderson, 1995). Unfortunately, language tutors have difficulty in developing strong models[1] of the student. Much of the difficulty comes from the inaccuracies inherent in automated speech recognition (ASR). Providing explicit feedback based only on student performance on one attempt at reading a word is not viable since the accuracy of the ASR at distinguishing correct from incorrect reading is not high enough (Williams, Nix, & Fairweather, 2000). Due to such problems, student modeling has not received as much attention in computer assisted language learning systems as in classic ITS (Heift & Schulze, 2003), although there are exceptions such as (Michaud, McCoy, & Stark, 2001).

[1] The research reported in this paper is applicable to both male and female students but, for convenience only, the student will be referenced as male.

Being able to model the student using ASR has implications beyond language tutors. For example, prior work (Pon-Barry, Clark, Schultz, Bratt, & Peters, 2004) has shown how properties intrinsic of speech input such as pauses, hesitations, and intonations provide a rich signal for adapting tutorial instruction. This signal is lacking from keyboard input. Other related work (Beck, Jia, & Mostow, 2004) showed that a student's pattern of hesitations, as measured by an ASR, were a better source of information for modeling his proficiencies than his help-requests. Clearly, speech input contains information of use to computer tutors; the bottleneck is determining how to extract useful information from a noisy signal.

Another reason for the lack of strong student models in language tutors is the disconnect between standard methods for developing student models and language learning (particularly reading). A common approach to developing cognitive models for use in an ITS is to use think-aloud protocols (Anderson, 1993; Newell & Simon, 1972). In a think-aloud study (Newell & Simon, 1972), participants verbalize their thinking while solving a problem. Such verbalizations are then used to construct a cognitive model of how the participants were solving the task. This approach has also been used to develop cognitive models for ITS (Anderson, 1993). Unfortunately, due to the speed of the reading process, think-aloud methodology is not well suited to modeling reading.

There have been efforts to develop cognitive models that describe the reading process. For example, (Harm, McCandliss, & Seidenberg, 2003) developed a parallel distributed processing model that was able to simulate many aspects of human performance. Unfortunately, their model is designed for individual word reading and not for reading connected text. Furthermore, the model is a psychological description of the reading process rather than a description of an individual's knowledge. Thus, these models are appropriate for examining effects such as why certain types of reading instruction are more effective than others (Harm et al., 2003). However, they do not provide (much) leverage for constructing models of an individual's reading proficiencies.

Our goal is to use speech recognition to reason about the student's proficiency at a finer grain-size. Even if it is not possible to provide immediate feedback for student mistakes, it may be possible to collect enough data over time to estimate a student's proficiency at various aspects of reading. Lessons learned about how to estimate student skill proficiencies in the domain of reading should transfer to other tutorial domains where the broad bandwidth provided by speech recognition could be used to better model student knowledge.

We conduct this research in the context of Project LISTEN's Reading Tutor (Mostow & Aist, 2001). The Reading Tutor is an intelligent tutor that listens to students read aloud with the goal of helping them learn how to read English. Target users are students in first through fourth grades (approximately 6- through 9-year olds). Students are shown one sentence (or fragment) at a time, and the Reading Tutor uses speech recognition technology to (try to) determine which words the student has read correctly or incorrectly. Much of the Reading Tutor's power comes from allowing children to request help and from detecting some mistakes that students make while reading. It does not have the strong reasoning about the user that distinguishes a classic intelligent tutoring system, although it does base some decisions, such as picking a story at an appropriate level of challenge, on the student's reading proficiency.

KNOWLEDGE TRACING

Knowledge tracing (Corbett & Anderson, 1995) is an approach for estimating the probability a student knows a skill given observations of him attempting to perform the skill. First we briefly discuss the parameters used in knowledge tracing, then we describe how to modify the approach to work with speech recognition. Speech recognition introduces two problems into the student modeling process. First, it is a noisy reflection of the student's performance. Second, it does not report performance at the grain size we are interested in analyzing.

Parameters in Knowledge Tracing

For each skill in the curriculum, there is a P(k) representing the probability the student knows the skill. Each skill also has two learning parameters:

- P(L0) is the initial probability a student knows a skill
- P(t) is the probability a student learns a skill given an opportunity

However, student performance is a noisy reflection of his underlying knowledge. Therefore, there are two performance parameters for each skill:

- P(slip) = P(incorrect | student knows skill), i.e., the probability a student gives an incorrect response even if he has mastered the skill. For example, hastily typing "32" instead of "23."
- P(guess) = P(correct | student doesn't know skill), i.e. the probability a student manages to generate a correct response even if he has not mastered the skill. For example, a student has a 50% chance of getting a true/false question correct even if he does not know the material.

With knowledge tracing, more attention is usually given to the knowledge parameters. However, for this research the performance parameters are key. Figure 1 shows the structure of the classic knowledge tracing model. The lighter, dotted, line represents a student making a guess; the darker, dashed, line represents a student making a slip.

When the tutor observes a student respond to a question either correctly or incorrectly, it uses the appropriate skill's performance parameters (to discount guesses and slips) to update its estimate of the student's knowledge. A fuller discussion of knowledge tracing is available in (Corbett & Anderson, 1995).

Accounting for Speech Recognizer Inaccuracies

Although knowledge tracing updates its estimate of the student's internal knowledge on the basis of observable actions, this approach is problematic with the Reading Tutor since the output of automated speech recognition (ASR) is far from trustworthy. Figure 2 shows how both student and ASR characteristics mediate observations of student performance. In standard knowledge tracing, there is no need for the intermediate nodes (enclosed in the box) or their transitions to the observed student performance. However, since our observations of the student are noisy, we need additional possible transitions. FA stands for the probability of a False Alarm and MD stands for the probability of Miscue Detection. A false alarm is when the student reads a word correctly but the word is rejected by the ASR; a detected mis-

cue is when the student misreads a word and it is scored as incorrect by the ASR. Classic knowledge tracing assumes a noiseless environment. In a noiseless input system, FA would be 0 and MD would be 1, and there would therefore be no need for the additional transitions. However, in the Reading Tutor, FA ≈ 0.04 and MD ≈ 0.25. The estimated rate of MD only counts cases where the student actively misread the word, as the tutor is much better at detecting skipped words that it is at detecting misreading.

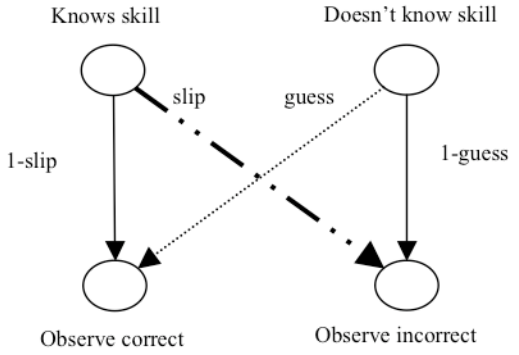

Fig.1. Structure of knowledge tracing's performance parameters.

All we are able to observe is whether the student's response is scored as being correct and the tutor's computed estimate of his knowledge based on past performance. Given these limitations, any path that takes the student from knowing a skill to generating an incorrect response is considered a slip. We denote these paths in Figure 2 the same way as in Figure 1 (lines denoting slips are thicker and dashed; lines denoting guesses are lighter and dotted). Note that there are two ways of going from knowing the skill to generating an observation of incorrect performance:

1. The student could accidentally slip, and the resulting miscue is detected by the ASR.
2. The student could read the word correctly, but the ASR has a false alarm and mistakenly scores the word as misread.

We define a variable called slip' to refer to either of these means of generating a slip. Mathematically slip' = slip * MD + (1-slip) * FA.

Similarly, a guess is any path from the student not knowing the skill to an observed correct performance. There are two ways a student could not know a skill but still generate an observation of correct performance:

1. The student could guess the correct reading of the word, and the ASR scores it as correct reading.
2. The student could fail to guess the correct reading of the word, but the ASR fails to detect the miscue and scores it as correct reading.

To refer to these two means of generating a guess, we create a parameter called guess' = guess * (1-FA) + (1-guess) * (1-MD).

In this framework, there is no need to consider or to explicitly model the ASR; the guess' and slip' parameters account for the variability it introduces. This approach of folding in the ASR noise into the performance parameter estimates for each skill is convenient both

practically and conceptually. Practically, we expect ASR performance to vary based on the words being read. Therefore, it is not appropriate to use the overall MD and FA rates to derive guess' and slip' parameters via a theoretical model. Due to the ASR's uneven performance, MD and FA will vary for each skill. Rather than trying to estimate slip, guess, MD, and FA for each skill, it is simpler to simply estimate guess' and slip' directly from the data (see Section on Parameter estimation).

Conceptually, we can think of guess' and slip' as just being the standard guess and slip parameters. Mathematically we can use those in the knowledge tracing equations without modification. However, note that the semantics of P(slip) and P(guess) change when using knowledge tracing in this manner. These parameters now model both the student and the method for scoring the student's performance. However, the application of knowledge tracing for the updating of student knowledge and the computational methods for estimating the learning and performance parameters remain unchanged. For simplicity, we henceforth refer to guess' and slip' as guess and slip.

METHOD FOR APPLYING KNOWLEDGE TRACING

We now describe how we applied knowledge tracing to our data. First we describe the data collected, next we describe the reading skills we modeled, then we describe how to determine which words the student attempted to read, and finally discuss the knowledge tracing parameter estimates.

Description of Data

Our data came from 284 students who used the Reading Tutor in the 2002-2003 school year. The students using the Reading Tutor were part of a controlled study of learning gains, so were pre- and post-tested on several reading tests. Students were administered the Woodcock Reading Mastery Test (Woodcock, 1998), the Test of Written Spelling (Larsen, Hammill, & Moats, 1999), the Gray Oral Reading Test (Wiederholt & Bryant, 1992), and the Test of Word Reading Efficiency (Torgesen, Wagner, & Rashotte, 1999). All of these tests are human administered and scored.

Students' usage of the Reading Tutor ranged from 27 seconds to 29 hours, with a mean of 8.6 hours and a median of 5.9 hours. The 27 seconds of usage was anomalous, as only four other users had less than one hour of usage.

While using the Reading Tutor, students read from 3 words to 35102. The mean number of words read was 8129 and the median was 5715. When students read a sentence, their speech was processed by the ASR and aligned against the sentence (Tam, Mostow, Beck, & Banerjee, 2003). This alignment scores each word of the sentence as either being accepted (heard by the ASR as read correctly), rejected (the ASR heard and aligned some other word), or skipped. In Table 1, the student was supposed to read "The dog ran behind a house." The bottom row of the table shows how the student's performance would be scored by the tutor.

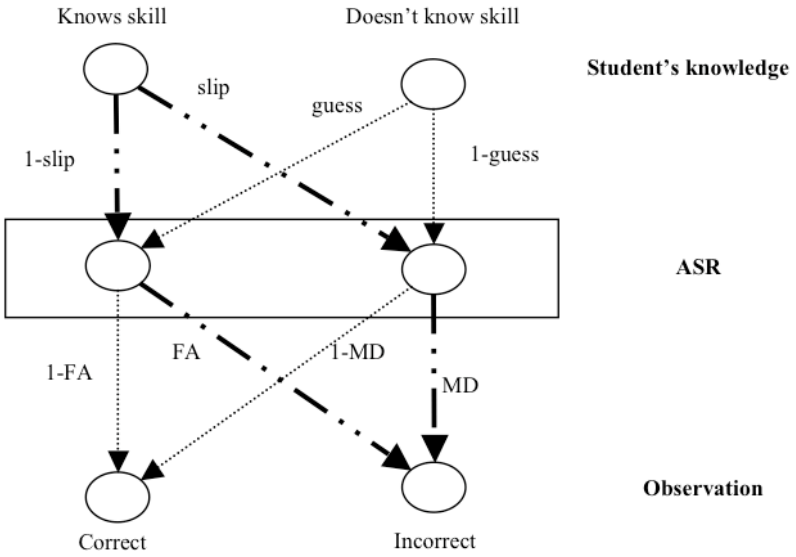

Fig.2. Knowledge tracing with imperfect scoring of student responses.

Table 1
Example alignment of ASR output to sentence

Sentence	The	dog	ran	Behind	a…
ASR output	The	the	ran		
Scoring	Accept	Reject	Accept	Skipped	Skipped

What Reading Skills to Assess?

Given the ASR's judgment of the student's reading, we must decide which reading skills we wish to assess. We could measure the student's competency on each word in the English language, but such a model would suffer from two major problems. First, due to most words being seen few times, it would be hard to obtain reliable estimates of the student's knowledge of many words. Second, students don't learn to read by memorizing the words of English as individual units; decoding knowledge about related words should transfer. Having a model that allows such transfer would also alleviate the sparse data problem. Therefore, we assess a student's knowledge of decoding as grapheme→phoneme (g→p) mappings. A grapheme is a group of letters in a word that produces a particular phoneme (sound). So our goal is to assess the student's knowledge of these g→p mappings. For example, the word "chemist" contains ch→/K/, e→/EH/, m→/M/, i→/IH/, s→/S/, and t→/T/ as g→p mappings.

We model g→p mappings since a particular grapheme can map to multiple phonemes. For example, *ch* can make the /CH/ sound as in the word "Charles." However, *ch* can also make the /K/ sound as in "chaos." By assessing students on the component skills necessary to read a word, we hope to build a model that will allow the tutor to make predictions about words the student has not yet seen. For example, if the student cannot read "chaos" then he

probably cannot read "chemistry" either, since each begin with the same g→p mapping. However, not being able to read "chemistry" says very little about the student's decoding knowledge of the word "Charles."

Modeling the student's proficiency at a subword level is difficult, as we do not have observations of the student attempting to read g→p mappings in isolation. There are two reasons for this lack. First, speech recognition is imperfect differentiating individual phonemes. Second, the primary goal of the 2002-2003 Reading Tutor is to have students learn to read by reading connected text, not to read isolated graphemes with the goal of allowing the tutor to assess their skills. To overcome this problem, we apply knowledge tracing to the individual g→p mappings that make up the particular word. However, which mappings are indicative of a student's skill? Prior research on children's reading shows that children are often able to decode the beginning and end of a word, but have problems with the interior (Perfetti, 1992). Therefore, we ignore the first and last g→p mappings of a word and use the student's performance reading a word to update the tutor's estimate of the student's knowledge of the interior g→p mappings. In the above example we would update the student's knowledge on e→/EH/, m→/M/, i→/IH/, and s→/S/. Words with fewer than three graphemes do not adjust the estimate of the student's knowledge.

Which Words to Score?

When students read a sentence in the Reading Tutor, sometimes they do not attempt to read all of the words in the sentence. If the student pauses in his reading, the ASR will score what the student has read so far. For example, in Table 1, the student appears to have become stuck on the word "behind" and stopped reading. It is reasonable to infer the student could not read the word "behind." However, the scoring of "a" and "house" depends on what skills are being assessed. If the goal is to measure the student's overall reading competency, then counting those words as read incorrectly will provide a better estimate since stronger readers will need to pause fewer times. Informal experiments on our data bear out this idea.

However, our goal is not to assess a student's *overall* reading proficiency, but to estimate his proficiency at particular g→p mappings. Since the student did not even try to read the words "a" and "house," they provide no information about the student's competency on the mappings that make up those words. Therefore we do not apply knowledge tracing to those words.

Informally, we compute the "zone" of the sentence the student was attempting to read. Words within the zone are scored according to the ASR's accept/reject decisions. Words outside the zone are ignored for purposes of student modeling.

More formally, we estimate the words a student attempted as follows:

1. i = Find the first word in the sentence that was accepted
2. j = Last word in the sentence that was accepted
3. Apply knowledge tracing to sentence words $i \ldots j+1$

In the example in Table 1, $i=1$ and $j=3$, and the words 1 through 4 would be scored ("The dog ran behind"). The reason the fourth word is scored is the student stopped reading for *some* reason. A plausible rationale for stopping, as is assumed by our heuristic, is because he could not read the next word in the sentence.

Parameter Estimation

We have described how to take the aligned ASR output and to use a heuristic to determine which words to score, and which g→p mappings in the words to model. We sorted each student's data chronologically, screened out words that were not in the zone, and scored the interior g→p mappings as either read correctly or incorrectly. Since we treat each g→p mapping as a distinct, transferable, skill, we combine all of a student's attempts at a particular g→p mapping across all words. We perform this procedure for each g→p mapping for each student.

There are 429 distinct g→p mappings that occur in at least one word in our dictionary. We use data from all students to estimate, for each g→p mapping, the four knowledge tracing parameters (L0, t, guess, slip)[2]. We then restricted the set of mappings to those with at least 1000 attempts combined from all students. We also removed mappings that fit the knowledge tracing model poorly. To test whether a g→p mapping was poorly modeled by knowledge tracing, we generated theoretical performance curves for a student who would behave according to the knowledge tracing model, and then compared the theoretical curve to the empirical data we collected on student performance. We excluded rules that had an R^2 of less than 0.20. These restrictions limited the set to 80 mappings.

The optimization code required some modification since it was designed for more traditional knowledge tracing. For example, the code restricted the number of "exercises" where students get to apply a particular skill to be less than 100. In our case, an exercise is a student attempting to read a word containing a particular g→p mapping. Some students encounter a particular mapping thousands of times. Another restriction is that P(guess) was forced to be less than 0.3 and P(slip) to be less than 0.1. We had to remove this restriction since mappings with at least 10,000 observations had an average P(guess) of 0.71 and P(slip) of 0.13.

The reason P(guess) is so high is that the Reading Tutor is biased towards hearing the student read the sentence correctly in order to reduce frustration from novices having correct reading scored as incorrect. These data demonstrate that with current speech recognition technology, a tutor cannot provide the same type of immediate feedback as a tutor with typed input due to the uncertainty in whether the student was correct. With such a high guess parameter, many observations are required for a student to be considered proficient in a skill. Fortunately, students read hundreds of words each day they use the Reading Tutor, so the bandwidth should be sufficient to estimate the student's proficiencies. As ASR quality improves, the performance parameters should decrease, and be more comparable to those obtained from classic, keyboard-driven, systems.

Once the above steps have been performed, we have a set of knowledge tracing parameter estimates for 80 g→p mappings. By taking the aligned output of the ASR of the student's reading, we can apply the knowledge tracing model to estimate the student's proficiency on each skill. This process results in a probability estimate as to whether the student knows each of the 80 reading skills in our model.

---

[2] Source code is courtesy of Albert Corbett and Ryan Baker and is available at http://www.cs.cmu.edu/~rsbaker/curvefit.tar.gz

VALIDATION

We now discuss validating our model of the student's reading proficiency. First we demonstrate that, overall, it is a good model of how well students can identify words. Then we show that the individual g→p estimates have predictive power.

Performance at Predicting Word ID Scores

If we run knowledge tracing over the student's Reading Tutor performance for the year, we get a set of 80 probabilities that estimate the student's proficiency at each g→p mapping. To validate the accuracy of these probabilities, we use them to predict the student's Word Identification (WI) post-test score from the Woodcock Reading Mastery Test (Woodcock, 1998). The post-test occurred near the end of the school year. For the WI test, a human presents words for the student to read and records whether the student read the word correctly or not, and terminates the test when the student gets four words in a row incorrect. The WI test is a good test for validating the overall accuracy of our g→p mappings since it presents students with a series of words; the student then either recognizes the word on sight or segments the words into graphemes and produces the appropriate phonemes. Since the student model is updated based on student's performance at reading words (as scored by the ASR), an external test where humans score the student's performance at reading words is a good match.

The goal is to use the estimates of the student's knowledge of the 80 g→p mappings to predict his grade equivalent WI post-test score. Grade equivalent scores are of the form *grade.month*, for example 3.4 corresponds to a third grader in the fourth month of school. The month portion ranges from 0 to 9, with summer months excluded.

Grade equivalent scores can be misleading. For example, a math test of simple addition may show that a first-grader had a score of 5.3. This result does **not** mean the student has the math proficiency of a fifth grader, rather it means that he scored as well as a fifth grader might be expected to do on that test (so the student is quite skilled at addition, but the score says nothing about his knowledge of other math skills a fifth grader would be expected to know, such as fractions).

In contrast, many reading tests are designed for grades K-12 (roughly ages 5 through 17). For example, in WI, the test starts with easy words such as "red" and "the." For a student to receive a score of 5.3, the student would have to read words such as "temporal" or "gruffly." If a first grader can read such words (and the preceding words on the test), it is not unreasonable to say he can identify words as well as a fifth grader (although his other reading skills may be lacking). As a target for building a model of the student, the grade equivalent scale is a reasonable choice due to its interpretability by researchers. This use of grade equivalent scores follows guidelines (Canadian Psychological Association, 1996) for when their use is appropriate.

We expect different g→p mappings to be predictive for students in different grades since skills that students have mastered in prior grades are unlikely to remain predictive in later grades. Therefore, we constructed a model for each grade. We entered terms into the regression model until the change in R^2 was less than 0.01 for grades one and two and less than 0.05 for grades three and four (there were fewer students in grades 3 and 4). This process

resulted in ten mappings entering the model for grade one, 25 mappings for grade two, five mappings for grade three, and four mappings for grade four.

The resulting regression model for WI scores had, using a leave-one-out cross validation, an overall correlation of 0.88 with the WI test. It is reasonable to conclude that our model of students' word identification abilities is in reasonable agreement with a well-validated instrument for measuring the skill. We examined the case where our model's error from the student's actual WI was greatest: a fourth grader whose pre-test WI score was 3.9, her post-test was 3.3, and our model's prediction was 6.1. It is unlikely the student's proficiency declined by 0.6 grade levels over the course of the year, and it was unclear whether we should believe the 3.3 or the 6.1. Perhaps our model is more trustworthy than the gold standard against which we validated it? There are a variety of reasons not to trust a single test measurement, including that it was administered on a particular day. Perhaps the student was feeling ill or did not take the test seriously? Also, we would like to know if our measure is better than WI. To get around these limitations, we looked at an alternate method of measuring word identification.

Alternate Measure of Word Identification

To find an alternate method of measuring word identification, we examined our battery of tests we administer to students to find a set of tests that are most similar to WI. The goal is to find a proxy for WI that does not suffer from the measurement errors described above. Then, we can see how well our student model predicts that proxy. To be a good proxy to WI, a test should have three characteristics:

1. It should correlate highly with WI. If a test only correlates weakly with WI, it would be odd to use it as a proxy for WI.
2. The test should measure the same underlying construct. It is possible for two measures to correlate strongly but to measure different constructs. For example, for grade school children shoe size and WI score are strongly correlated: older children can generally read better and have larger feet. However, shoe size has nothing to do with WI, and it would not be appropriate to use it to replace WI.
3. The measurement error on the proxy should be relatively unrelated to the measurement error on the WI test. If the proxy measures the same construct, is correlated with WI, but has the same measurement error, it is not providing any additional information.

The WI test is not an appropriate choice to use as a proxy for two reasons. First, its measurement error is of course identical to the measurement error in the WI test, so it is not a useful source of information. Second, we would like to compare how the student model and WI do at predicting the proxy for WI; if WI is a member of the composite proxy then the comparison would not be fair.

The three best tests were:

1. The Accuracy score from the Gray Oral Reading Test (GORT) measures how many mistakes students make reading connected text. It correlates with WI at 0.76.
2. Sight Word Efficiency (SWE) from Test Of Word Reading Efficiency (TOWRE) measures how quickly students can decode common words. It correlates with WI at 0.80.

3. The Test of Written Spelling (TWS) is the opposite of word identification as students are presented a sound and asked to generate the proper letters, but is related to word identification (Carver, 2003) and correlates with WI at 0.86.

None of these measures perfectly matches the construct of word identification, but they measure closely related constructs. We took the mean of these three tests as a proxy for the student's word identification proficiency. An advantage of taking the mean of three tests is that if a student has a high measurement error on one of the tests (perhaps due to misunderstanding the task or to not enjoying the format), the error will be somewhat ameliorated by the other measures. Furthermore, since these tests were sometimes administered on different days and usually by different testers, negative effects due to not feeling well on a particular day or not liking a particular tester are reduced. The mean of the three tests correlates with WI at 0.87. Furthermore, the mean of the 3 scores (hereafter called WI3) does not suffer nearly as badly as WI from students dropping several months in proficiency from pre- to post-test. Given the stability of the WI3 measure, its being composed of constructs closely related to word identification, and its statistical correlation with WI, we feel it is a good measure of the students "true" word identification score.

Returning to the student whose WI post-test score deviated from the model. Her WI post-test score was 3.3, her score as predicted by the student model was 6.1, and her WI3 score was 5.1. Perhaps our model did a better job for assessing this student's word identification proficiency than the WI test? To evaluate the accuracy of our model, we compared our model and the WI score to see how often each was closer to the WI3 score. The WI test was closer to the WI3 score 52.8% of the time, while our model was closer 47.2% of the time. An alternate evaluation is to examine the mean absolute error (MAE) between each estimate and WI3. WI had an MAE of 0.71 (SD of 0.56), while our model had an MAE of 0.77 (SD of 0.67), a difference of only 0.06 GE (roughly three weeks). So our model was marginally worse than the WI test at assessing (a proxy for) a student's word identification abilities. However, the WI test is a well-validated instrument, and to come within 0.06 GE of it is an accomplishment. Although marginally worse than the paper test, the knowledge tracing model can estimate the student's proficiency at any time throughout the school year, requires no student time to generate an assessment, and does not require trained testers to administer.

Predicting Help Requests

To validate whether our model's estimates of the student's knowledge of individual g→p mappings were accurate, we predicted whether the student would ask for help on a word. We used help requests rather than the student's performance at reading words since we already extracted considerable data about student reading performance to build our model. Thus using it to confirm our model would be circular.

To measure whether knowledge of g→p mappings would be predictive of whether the student would ask for help, we examined every word the student encountered and noted whether he asked for help or not. We excluded words composed of fewer than three graphemes (since our model is based on student performance on interior g→p mappings). Approximately 79% of English tokens in children's reading materials are composed of 3 or more graphemes. The above restrictions limited us to 288,614 sentence word tokens the students encountered during their time using the Reading Tutor.

We constructed a logistic regression model to predict whether a student would ask for help on a word. This model had several components:

1. The identity of the student was a factor. Adding the student to the model controls for overall student ability and for student differences in help-seeking behavior (in the past, student help request rates have differed by two orders of magnitude in the Reading Tutor).
2. The difficulty of the word (on a grade equivalent scale) was a covariate. Presumably students are more likely to ask for help on difficult words.
3. The position of the word in the sentence was a covariate. In the Reading Tutor, students sometimes do not read the entire sentence. Therefore, we suspected that words earlier in the sentence are more likely to be clicked on for help.
4. The average knowledge of the 80 g→p for the student at the point in time when he encountered the word was a covariate. This term modeled the changes in the student's knowledge over the course of the year.
5. The student's average knowledge of the g→p mappings that composed the word, excluding the first and last mappings, was a covariate in the model. For words with 3 or more graphemes, the modal number of graphemes was 3 and the median was 4. Therefore, there are generally only one or two interior g→p mappings, so the student's average knowledge of the mappings in a word was not a broad description of the student's competencies, but is a focused description of his knowledge of the components of this word.

Logistic regression generates Beta coefficients to determine each variable's influence on the outcome, assuming that the other model terms are held constant. Factors refers to variables that are an ordinal scale; logistic regression estimates a Beta for each value the factor can take on. Therefore, there is no overall Beta value for the student identity, rather each student had a separate Beta value. The purpose in using student identity as a factor is to control for the fact that students have varying numbers of trials and to allow us to obtain p-values from the model that do not over- or under-count N (Menard, 1995). Covariates refer to variables where a linear influence is assumed; for such variables Beta refers to the increase in the chance the student will ask for help for each unit the covariate increases. Beta values are not normalized; i.e. if a covariate is multiplied by 10 then its Beta values will be 10 times smaller. Therefore, we cannot compare the covariate's Betas to see which of them is more influential.

The Beta coefficients were 0.48 for word difficulty, -0.035 for the word's position in the sentence, -0.96 for the student's mean proficiency on the 80 g→p mappings, and -0.38 for the student's mean proficiency of the interior g→p mappings in the word, and. If a variable has a positive Beta coefficient, then as the variable's value increases the student's probability of asking for help increases. Conversely, a negative Beta implies as the value increases, the student's probability of requesting help decreases. All of the Beta values were significant at $P<0.001$, and all point in the intuitive direction: students ask for help on harder words, are less likely to ask for help at the end of a sentence, and stronger readers are less likely to request help. Since the Beta values represent the influence a variable has while holding the other variables constant, this means that even when student identity and word difficulty are held constant, the student's proficiency on the interior g→p mappings is a significant predictor of his likelihood of requesting help.

These results provide evidence that individual estimates of the student's proficiency on g→p mappings are meaningful indicators of proficiency.

CONCLUSIONS AND FUTURE WORK

This paper demonstrates that it is possible to use speech recognition for the purposes of student modeling by applying classic student modeling techniques. This result was not entirely expected, and there was considerable skepticism at the start of the enterprise. While it is true the ASR data are extremely noisy, it is possible to account for the noise and model student proficiency on subword skills, in our case g→p mappings, of reading. This model of proficiency is accurate in the aggregate since it is able to assess a student's word identification proficiency nearly as well as a paper test designed for the task. Furthermore, the individual estimates of the student's knowledge are also useful, since they predict whether a student requests help on a word even after controlling for his overall reading proficiency.

Although the student model was not quite as accurate as the Word Identification test to which it was being compared, the student model approach has two advantages. First, the student model can be queried at any point in the year to get an estimate of the student's reading proficiency; the paper test must be administered for each such assessment. Second, the student modeling approach takes no time away from the student's instructional time. The natural process of working with the Reading Tutor provides sufficient data to assess the student.

This work extends the state of the art in using speech recognition to assess students. The most similar prior work is Ordinate's SET10 test of spoken English. The SET10 uses automated speech recognition to assess student's proficiency at understanding and responding to simple English questions. The SET10 requires students to respond to scripted situations, as opposed to the work presented here which takes natural observations of students reading aloud. Furthermore, the SET10 only produces a composite score of the student's proficiency. Our work not only models student overall performance, but also assesses students at a fine grain size.

Two other efforts similar to Project LISTEN's Reading Tutor that also assess students using speech recognition are IBM's Watch Me! Read project[3] (WM!R) and the University of Colorado's Reading Tutor project[4]. WM!R has done work on assessing students (Williams et al., 2000); the system provided teachers with passages the student had read as well as which words the speech recognizer thought the student read incorrectly. However, this work was formative, and due to concerns about speech recognizer inaccuracy, WM!R did not attempt to distill its word-level scoring into an overall estimate of student reading proficiency (Williams, 2002). The University of Colorado's Reading Tutor Project also assesses student reading skills including reading comprehension. However, these assessments are not generated via the student's natural reading of text. To assess student knowledge of skills, the system uses computer administered instruments. To assess comprehension, students type a summary of the text they have read and the system uses latent semantic analysis (Deerwester, Dumais, Furnas, Laundauer, & Harshman, 1990) to analyze the accuracy of the student's summary (Wade-Stein & Kintsch, 2004).

---

[3] http://www.ibm.com/ibm/ibmgives/grant/education/programs/reinventing/watch.shtml
[4] http://cslr.colorado.edu/beginweb/reading/reading.html

There are several avenues for expanding the scope of the assessment work presented in this paper: a better credit model for scoring student performance, using cues other than ASR acceptance/rejection, having a richer cognitive model than the simple g→p model, and using the derived student model to enhance the performance of the speech recognizer.

Our approach for constructing a student model from the ASR output is somewhat crude. Currently, all of the g→p mappings in a word are blamed or credited. However, this mechanism has two shortcomings. First, if a student misreads a word it is probable that not all of the mappings are responsible. A Bayesian credit assignment approach (e.g. (Conati et al., 2002)) would overcome this weakness. Second, not all g→p mappings in a word have the same probability of causing a student to misread the word. For example, if the student doesn't know the last mapping in a word he may be able to guess at how the word ends, while a student who does not know the first mapping of a word may be unable to decode it even if he knows the remaining mappings.

One method of accounting for this problem would be to have a Bayesian network with a structure that encodes positional information. For example, a network could have 3 proficiency nodes that link to observed performance. The first node would represent the student's proficiency on the first g→p mapping, the second could be the mean of all interior g→p mappings, and the third node could represent the proficiency on the final g→p mapping. We expect the performance node would more heavily depend on the first g→p than on the other two nodes. An alternate model is to create a separate Bayesian network for words of each number of g→p mappings. Either of these approaches of accounting for the word's structure should enable better modeling.

This work also uses a rather naïve model of student development of reading skills. While students certainly make use of letter→sound mappings while learning how to read, there are other plausible representations. For example, as students become more familiar with a word, they may transition from using the g→p mappings that compose the word to directly accessing the word from memory as a whole unit. We have performed initial experiments (Chang, Beck, Mostow, & Corbett, 2005) that demonstrate that a combination of whole-word and g→p models are better able to model students. However, much work remains to be done in this area.

The student's pattern of hesitation before a word contains a useful signal for modeling the student (Beck et al., 2004; Mostow & Aist, 1997). One possible future avenue is to use the amount of hesitation before reading a word as a clue to the strategy the student is using: a short pause suggests a whole word strategy while a longer pause suggests the student is using his knowledge of g→p mappings. This signal could disambiguate between which strategy a student is using to decode, which is important information about the student's progress in learning how to read.

Another area of future exploration is using the student model to improve the ASR. Part of the reason constructing a student model from an ASR is difficult is the statistical noise introduced by the speech recognition. If the student model can provide clues to the ASR about how to better listen to the student, then the recognition could be improved. This improvement in recognition would make the student modeling task easier, which could result in an improved student model. One approach is to use the student model to second guess the speech recognizer. For example, if the student model believes a student can read a word correctly, but the ASR hears the word as being misread, the student model could overrule the ASR and disregard the student error. Initial experiments (Beck, Chang, Mostow, & Corbett, 2005) at using the student model and ASR to predict student reading (as judged by a human

transcriber) demonstrated a significant improvement in the area under curve metric for classifier sensitivity than compared to just using ASR. Determining how to better combine the student model with the ASR, and how their conflicting judgments should be resolved is an open issue.

ACKNOWLEDGEMENTS

This work was supported by the National Science Foundation, ITR/IERI Grant No. REC-0326153. Any opinions, findings, and conclusions or recommendations expressed in this publication are those of the authors and do not necessarily reflect the views of the National Science Foundation or the official policies, either expressed or implied, of the sponsors or of the United States Government. We also acknowledge members of Project LISTEN who contributed to the design and development of the Reading Tutor, and the students who used the tutor.

REFERENCES

Anderson, J. R. (1993). *Rules of the Mind*: Lawrence Erlbaum Assoc.

Beck, J. E., Chang, K-M., Mostow, J., & Corbett, A. (2005). Using a student model to improve a computer tutor's speech recognition. In S. Alpert & J. E. Beck (Eds.) *Student modeling for language tutors workshop at International Conference of Artificial Intelligence and Education* (pp. 2-11). Amsterdam.

Beck, J. E., Jia, P., & Mostow, J. (2004). Automatically assessing oral reading fluency in a computer tutor that listens. *Technology, Instruction, Cognition and Learning*, 2, 61-81.

Canadian Psychological Association (1996). *Guidelines for Educational and Psychological Testing*. Also available at: http://www.acposb.on.ca/test.htm.

Carver, R. P. (2003). The highly lawful relationship among pseudoword decoding, word identification, spelling, listening, and reading. *Scientific Studies of Reading,* 7(2), 127-154.

Chang, K.-M., Beck, J. E., Mostow, J., & Corbett, A. (2005). Using Speech Recognition to Construct a Student Model for a Reading Tutor. In S. Alpert & J. E. Beck (Eds.) *Student modeling for language tutors workshop at International Conference of Artificial Intelligence and Education* (pp. 12-21). Amsterdam.

Conati, C., Gertner, A., & VanLehn, K. (2002). Using Bayesian Networks to Manage Uncertainty in Student Modeling. *User Modeling and User-Adapted Interaction*, 12(4), 371-417.

Corbett, A. T., & Anderson, J. R. (1995). Knowledge tracing: Modeling the acquisition of procedural knowledge. *User Modeling and User-Adapted Interaction,* 4, 253-278.

Deerwester, S., Dumais, S. T., Furnas, G. W., Laundauer, T. K., & Harshman, R. (1990). Indexing by Latent Semantic Analysis. *Journal of the American Society for Information*, 41, 391-407.

Harm, M. W., McCandliss, B. D., & Seidenberg, M. S. (2003). Modeling the successes and failures of interventions for disabled readers. *Scientific Studies of Reading*, 7(2), 155-182.

Heift, T., & Schulze, M. (2003). Student Modeling and ab initio Language Learning. *System, the International Journal of Educational Technology and Language Learning Systems,* 31(4), 519-535.

Larsen, S. C., Hammill, D. D., & Moats, L. C. (1999). *Test of Written Spelling* (Fourth Ed.). Austin, Texas: Pro-Ed.

Menard, S. (1995). Applied Logistic Regression Analysis. *Quantitative Applications in the Social Sciences*, 106.

Michaud, L. N., McCoy, K. F., & Stark, L. A. (2001). Modeling the Acquisition of English: an Intelligent CALL Approach. *8th International Conference on User Modeling* (pp. 14-23). Berlin: Springer.

Mostow, J., & Aist, G. (1997, July). The Sounds of Silence: Towards Automated Evaluation of Student Learning in a Reading Tutor that Listens. *Proceedings of the Fourteenth National Conference on Artificial Intelligence (AAAI-97)* (pp. 355-361). Providence, RI: American Association for Artificial Intelligence.

Mostow, J., & Aist, G. (2001). Evaluating tutors that listen: An overview of Project LISTEN. In K. Forbus & P. Feltovich (Eds.), *Smart Machines in Education* (pp. 169-234). Menlo Park, CA: MIT/AAAI Press.

Newell, A., & Simon, H. (1972). *Human Problem Solving*. Englewood Cliffs, NJ: Prentice-Hall.

Perfetti, C. A. (1992). The representation problem in reading acquisition. In P. B. Gough & L. C. Ehri & R. Treiman (Eds.), *Reading Acquisition* (pp. 145-174). Hillsdale, NJ: Lawrence Erlbaum.

Pon-Barry, H., Clark, B., Schultz, K., Bratt, E. O., & Peters, S. (2004). Advantages of Spoken Language Interaction in Dialogue-based Intelligent Tutoring Systems. In J. C. Lester, R. M. Vicari & F. Paraguaçu (Eds.) *Intellligent Tutoring Systems* (pp. 390-400). Berlin: Springer.

Tam, Y.-C., Mostow, J., Beck, J., & Banerjee, S. (2003, September 1-4). Training a Confidence Measure for a Reading Tutor that Listens. *Proc. 8th European Conference on Speech Communication and Technology (Eurospeech 2003)* (pp. 3161-3164). Geneva, Switzerland.

Torgesen, J. K., Wagner, R. K., & Rashotte, C. A. (1999). *TOWRE: Test of Word Reading Efficiency*. Austin: Pro-Ed.

Wade-Stein, D., & Kintsch, E. (2004). Summary Street: Interactive computer support for writing. *Discourse Processes*, 22, 333-362.

Wiederholt, J. L., & Bryant, B. R. (1992). *Gray Oral Reading Tests* (3rd ed.). Austin, TX: Pro-Ed.

Williams, S. M. (2002). Speech recognition technology and the assessment of beginning readers. In NRC (Ed.), *Technology and assessment: Thinking ahead: Proceedings of a workshop* (pp. 40-49). Board on Testing and Assessment, Center for Education, Division of Behavioral and Social Sciences and Education. Washington, DC: National Academy Press.

Williams, S. M., Nix, D., & Fairweather, P. (2000). Using Speech Recognition Technology to Enhance Literacy Instruction for Emerging Readers. In B. Fishman & S. O'Connnor-Divelbiss (Eds.) *Fourth International Conference of the Learning Sciences* (pp. 115-120). Lawrence Erlbaum.

Woodcock, R. W. (1998). *Woodcock Reading Mastery Tests - Revised (WRMT-R/NU)*. Circle Pines, Minnesota: American Guidance Service.

Woolf, B. P. (1992). AI in Education. *Encyclopedia of Artificial Intelligence* (Second Ed., pp. 434-444). New York: John Wiley & Sons.