
Personalized Learning in the Elementary School ELA Classroom

Dr. Dan Flickinger, Stanford University

Instructional software for the English Language Arts classroom has been available to teachers and students for decades, often as on-line implementations of conventional printed workbook-style practice materials. Surprisingly few of these supplementary tools incorporate an adaptive capability which would enable the software to change the pace or order in which concepts are presented, or to modify its guidance for each student, based on the student's responses to exercises presented by the software system.

Even fewer available software tools, adaptive or not, provide instruction and practice in writing, with most focusing instead on improving reading and vocabulary skills. This paper presents a rationale for developing software for ELA instruction which includes in its core design the ability to personalize the behavior and responses of the system in order to align with each student's pace of learning, with particular focus on instruction and practice in writing.

Adaptive vs. Linear Supplemental Software

The concept of incorporating an adaptive approach in education software goes back more than 50 years, pioneered by Patrick Suppes, Ph.D., at Stanford University—first for mathematics and much later for ELA. Suppes designed and implemented algorithms to enable an educational software system to change its behavior in at least two important ways in response to a student's responses: one to determine whether and when a student has gained sufficient proficiency with a concept in order to move onward, and the other to decide which concept to present next to the student. For individual concept proficiency, the system first introduces the concept with a brief tutorial, and then offers a set of exercises to give the student practice with the concept, observing the pattern of correct and incorrect answers to determine whether and when the student clearly understands the concept. The system employs a second algorithm to choose the next concept to present to the student—moving to a new, often more complex concept when the student showed proficiency, or to a precursor concept when the student had sustained difficulty with the exercises, and would benefit from review of earlier material.

Advantages of such adaptivity include: more effective use of the student's time in front of the computer, better engagement of the student's interest over a sustained period of using the software, and reduced frustration due to working with concepts that are either too easy or too difficult. With an adaptive system, students set their own pace for working through the concepts covered by the software system, moving more quickly through material that is either already familiar or readily learned, and taking more time with concepts that are challenging or require a review of prior lessons.

In contrast, a more conventional workbook-style software tool either prescribes for every student the same linear order of concepts and density of practice exercises, or expects the student to also play the role of instructor, choosing which concept to work on next, and for how long. It is clear that the uniform ordering and density can lead to frustration both for quick learners and for students who find the material challenging, but teachers can also confirm that many students do not have the perspective or the motivation to maximize the benefits of a tool, if they have to choose their own path alone. These choices are even more challenging for ELA than for more structured subjects such as mathematics, since there are many partly-independent strands of material to learn in parallel—including vocabulary building, phrase and sentence structure, paragraph structure, and reading, among others.

In one session with a supplementary software system, a student would ideally learn and practice concepts in several of these strands, improving skills in both writing and reading, but it is a lot to ask of students—that they make wise choices about how to allocate their time and effort—when they don't always know which concepts depend on having learned others first. Moreover, students differ in how readily they learn in each of these aspects or strands of ELA, and if asked to choose, will often spend more time doing what they already do well, neglecting concepts in other strands that also need to be learned.

For these non-adaptive educational tools, it is also in principle possible to expect the teacher to supply the expertise and discipline needed to make most effective use of the software and the students' time. However, this individual alignment of student to appropriate concept would need to be made not only when the student is first placed to start using the system, but at frequent intervals throughout its use, based on the teacher's observation of each student's rate of progress with the software's materials. Such constant manual tuning for a classroom of students will often impose impractical demands on the teacher's time, detracting from the potential benefits of using supplemental software. The resulting lack of individual tuning will leave many students frustrated at facing material in the system that is not suitable for their pace of learning.

Adaptive ELA in Use in the Classroom

One current embodiment of Suppes' adaptive algorithms in software for ELA is McGraw-Hill Education's supplemental Redbird Language Arts & Writing which addresses standards for grades 2-7, and is an expanded version of the language arts and writing course developed by Suppes and colleagues in Stanford's Education Program for Gifted Youth (EPGY). The units that make up the course include instructional materials which present each of the hundreds of concepts students need to learn in these grades, and with each concept comes a set of automatically evaluated exercises to help students learn and develop proficiency. The concepts and associated exercise sets include instruction and practice at the levels of word, phrase, sentence, and paragraph, with primary focus on developing writing skills, with the inclusion of reading lessons to provide models of good writing in a wide variety of genres and styles.

As outlined above, the software system observes the student's pattern of correct and incorrect responses to the exercises in each set, to determine based on one of Suppes' algorithms if and when the student has reached a suitable level of proficiency. Many of the exercises in these sets are essentially multiple-choice, readily evaluated automatically, but throughout the course students are also presented with writing exercises, both for one-sentence responses and for paragraph composition. Student responses to these exercises, even with limited vocabulary sets for each exercise, can show much more variation, both for correct and for incorrect answers, and thus a rather powerful evaluation engine is used to check for grammaticality and correct meaning.

Personalization in Responses to Writing

The evaluation software engine for composition exercises in this system is called a parser, and makes use of a large implemented grammar of English, also developed at Stanford over the past 25 years (Flickinger 2000, 2011), which has been extended to include a set of task-specific grammar rules that allow for specific types of errors frequently encountered in students' writing (Bender et al, 2004). When one of these so-called “mal-rules” is used by the parser in analyzing a sentence submitted by a student, the system presents the student with an associated error-specific message identifying the grammatical error and recommending a suitable correction. For example, a student's answer for an exercise might be “She wrote story about butterflies.” and the parser's analysis of this sentence will discover that the phrase “story about butterflies” is missing an article, because full analysis of the sentence will require use of one of the “mal-rules” added to the standard grammar—namely the rule that allows “story about butterflies” to be treated as a full object noun phrase, even though the obligatory article “a” or “the” has been omitted. Because this particular mal-rule had to be used by the parser, the system can present the student with the associated specific error message about the missing article, and point to the position in the sentence where it should be inserted.

Most of the single-sentence composition exercises in the course take the form of a set-up statement or two, followed by a question which the student answers by writing a complete sentence using a relatively small vocabulary list specific to that exercise. A second type of single-sentence exercise provides a prompt in the form of structured information presented as a table, chart, or graph, together with a question which requires the student to first interpret the information, and then compose a sentence that answers the question using the available vocabulary. For each of these exercises, a small set of reference correct answers is used by the system to further evaluate the student's answering sentence for correct meaning when it is grammatically well-formed.

Examples of frequently occurring grammatical errors addressed by the system include agreement in number between the subject of the sentence and the predicate (as in correct “The ducks swim” vs. incorrect “The ducks swims”), choice of pronoun case (“we” vs. “us”), and the inclusion of “to” preceding verb phrases (correct “They tried to swim” vs. incorrect “They tried swim”, but correct “They must swim” vs. incorrect “They must to swim”). The grammar used in the course currently identifies more than a hundred grammatical error types that occur frequently in the roughly 10 million sentences collected from students using the course over the past nine years.

Errors in the meaning of individual sentences are classified into broader categories, identifying incorrect choice of the subject or object of a sentence, or the omission of one or the other, or the unwanted inclusion of either. For example, if a correct answer to an exercise is “The cat chased the duck”, and the student's answer is “The cat was chased by the duck”, the system's response will advise the student that this answer, while perfectly grammatical, has the wrong subject.

If a student makes a mistake in writing a sentence, whether an error of grammar or of meaning, the system provides the appropriate error message and offers one chance to correct and resubmit the sentence for a second evaluation, before moving on to the next exercise. The system's error messages are specific to the particular words used by the student in the submitted sentence, where possible identifying the exact location of the error within the sentence, along with advice on the repair that is needed. This level of fine-grained, error-specific response is another important aspect of personalization, showing that the system is paying close attention to what the student has written, and offering advice on just how to improve that specific composition.

Responses in paragraph composition

In addition to the thousand single-sentence exercises woven through the course, students also receive instruction and practice in writing paragraphs of several types, in a scaffolded approach first offering practice composing a lead sentence, a supporting sentence, and a closing sentence, before presenting the challenge of writing a full paragraph from start to finish. The types of paragraphs taught and practiced in the course are narrative, informational, opinion, and persuasive or argumentative, starting with the first three types in the earlier grades, and adding persuasive/argumentative paragraph writing later in the course.

The system again provides detailed feedback evaluating the grammaticality of each sentence as it is submitted, and giving the student a chance to correct errors before adding the next sentence. In contrast to the single-sentence exercises, evaluation of correct meaning is not practical, since the range of variation in the structure of a paragraph is far greater than for individual sentences, even when providing the student with a relatively modest vocabulary list to draw from for each exercise.

Because these full-paragraph writing exercises are of necessity more time-consuming, the course includes fewer of these by far than of the single-sentence composition exercises. However, they provide the student with an important opportunity to gain practice and confidence in expressing more complex ideas that take several sentences to communicate, and they also provide the system and the teacher with a compact and informative view into where a student is succeeding or having difficulty in developing writing skills.

Beyond the paragraph

In 2018, the Redbird Language Arts & Writing course will provide students with an even more challenging and useful opportunity to expand a student's writing skills, in the form of an essay project, or Writing Review, that leads the student through the process of composing a multiple-paragraph essay. Building on the existing course instruction for individual paragraphs, the essay project provides additional scaffolding to guide the student in structuring this larger composition, and in working through the production step by step, while allowing significant expressive freedom. This project can give the student an opportunity to synthesize much of what they have learned, and apply that learning to create a more complex and interesting writing effort without the constraints of vocabulary and concept-specific structure that are a necessary part of the main Redbird Language Arts & Writing curriculum.

The course also includes language-learning games that give students opportunities to practice what they have learned so far in an engaging setting outside of the tutorial-and-exercise framework of the course. The benefits of these games go beyond engagement, important though that is, for these games also enable students to develop what Suppes called "automaticity," the kind of fluency with a newly acquired skill that comes only from so much practice that, here, correct usage of grammar and vocabulary becomes almost automatic. In one of these games, for example, the student needs to write short sentences in order to advance toward the goal, where writing more quickly (and correctly) is beneficial in this time-bounded context.

Efficacy of Adaptive Learning

In order to measure the impact of adaptive methods in software used in the classroom, Suppes and colleagues conducted a multi-year study (Suppes et al, 2014) with thousands of students in Memphis, Tennessee, providing them with supplemental course software both in mathematics and in English language arts that implemented the adaptive methods described above. The study recorded student scores on the standard end-of-year state examination for the year before the study began, and again at the end of each year of the study, which for the language arts course was a two-year span. The study also measured the level of effort of each student using the software by recording the amount of "positive" work, namely the percentage of correctly answered exercises minus the percentage of incorrectly answered ones.

By correlating the change in each student's year-over-year state exam scores with the amount of work the student did using the software during each year, the study found a highly significant effect size showing that exam scores increased as the amount of work increased, for students at all levels of proficiency, and for both years of the study. One further striking outcome of the study was that work with the software was most beneficial for students with the lowest initial proficiency in language arts—the lower the initial exam score, the greater the improvement in that score from one year to the next for students who invested the effort in the use of the software.

What the study shows is that positive effort by students using this adaptive software in the classroom correlates well with improved proficiency in language arts, as measured by performance on standard state examinations. For such a multi-year study to succeed, the students needed to remain engaged and willing to proceed through the course materials over the two-year period, and the instructional responses of the system needed to have a sustained positive effect on proficiency. The adaptive methods implemented in the software contributed to both of these needs, presenting each student with instruction and exercises well-matched to the individual's pace of learning, and giving specific feedback enabling the student to correct errors and improve proficiency concept by concept.

Ongoing Research in Personalized Learning at Stanford

The legacy of Suppes' work in adaptive methods for education continues at Stanford in research funded in part by McGraw-Hill Education, aimed at extending the capabilities of the software used in the Redbird Language Arts & Writing course.

Current work at Stanford's Center for the Study of Language and Information (CSLI) is focused in four areas:

- automatic analysis of short student essays for errors of both grammar and style
- automatic evaluation of short written answers to questions about reading passages
- adaptation of the evaluation engine for learners of English as a second language
- making use of the individual history of a student's answers in providing more personalized responses

More practice in writing short essays

Teachers of ELA broadly agree that students would benefit from more practice writing short essays, but the time demands of grading each of these compositions for errors of grammar, content, and style make it impractical to give such assignments with the desired frequency. What might free teachers to make essay-writing assignments more frequently is a software tool that automatically and accurately analyzes a student essay for errors of grammar and style, giving the student specific feedback to enable correction of most of these errors before submission.

The teacher could then spend much less time per essay, and focus on noting strengths and weaknesses in clarity, audience, interest, or truth of the content, all aspects of the writing which cannot be evaluated automatically. Even when a teacher would not find the time to give feedback on each essay, the student could find the grammar and stylistic error feedback motivation enough to stay engaged in the writing process, willing to continue to practice and thus improve proficiency.

For this approach to succeed, it is crucially important that the evaluation software is accurate in identifying a wide variety of error types in student writing, with sufficiently detailed analysis of the structure of each sentence to enable detailed guidance on how to make the necessary repairs. The Stanford team's approach is to extend the existing parser and grammar, used to evaluate Redbird Language Arts & Writing exercises with limited vocabulary, to accommodate open vocabulary and the wider range of expressive variation found in freely-written student essays.

Writing about reading

Even in a course focused on improving writing skills, it is important to include instruction and exercises in building reading skills, in order to provide models of good writing. The Redbird Language Arts & Writing course presents students with a variety of short reading passages at each grade level, both fiction and non-fiction, and provides guidance and practice related to vocabulary, writing style, audience, and purpose. These exercises about the reading passages currently do not yet include opportunities for the students to compose sentences in response, so the Stanford team is developing an extension of the evaluation engine's capabilities to enable students to write out answers to questions about a passage's content.

The approach being pursued makes use of a paraphrasing capability which can automatically produce from a small number of human-supplied reference answers a wide variety of sentences that express essentially the same meaning differently. If this rephrasing capability can be made general enough, the author of each exercise for a reading passage can supply a relatively small number of sentences that are each a correct answer to the question, and then the engine can generate a large number of additional forms of the correct answers. With this much larger set of correct target answers available, the system can be much more successful in evaluating whichever answer a student writes, and either confirm that the answer is correct, or give the necessary error-specific feedback for incorrect answers.

Adapting error analysis for second language learners

Many students in most school districts in the U.S. come from households where English is not the first language, and as second-language learners they often make errors when composing English sentences that are influenced by characteristics of their first language. For example, native speakers of Spanish may more readily leave out the subject when writing an English sentence, since subjects are normally optional in Spanish. They might also more readily reverse the order of subjects and objects, since unlike English, objects in Spanish can come before the verb in certain contexts. If the evaluation engine analyzing these sentences has information about the first language of the student, it could in principle provide a more informative error message that takes into account the differences between that language and English.

The Stanford team is approaching this task by working first with Mandarin Chinese and with Spanish, in each case identifying salient contrasts in their grammatical systems compared with English, and adding language-specific “mal-rules” and error messages that are informed by these contrasts. Pilot studies with students will provide data to guide the development of extensions for error types that occur frequently, and enable an initial study of how effective these more personalized error messages are in helping second-language English learners develop proficiency in writing.

Using student history for better personalization of responses

As students work through the Redbird Language Arts & Writing course, the system accumulates a rich storehouse of information about their learning paths, including a record of when and how well each concept was learned, and how consistently that proficiency has been demonstrated over time. If the evaluation engine for student writing had access to this historical record for each student, it would be possible to give the student more nuanced responses, including positive feedback such as “Nice work with that relative clause! You haven’t been using them very often, so it’s good to see this one.” For a concept that the student occasionally forgets, the system could respond as follows: “Remember what you learned two weeks ago about pronouns with the word ‘and’—We don’t write ‘Me and Sally rode the bus’ but rather ‘Sally and I rode the bus.’ Let’s practice this again.”

While this more personalized level of response would also require some nontrivial additions to the general course software system used for Redbird Language Arts & Writing and other subjects, the Stanford team is developing a prototype that assumes the availability of some types of historical data for each student, in order to explore candidate response types which could be generated automatically. At the least, such enriched responses should improve engagement with the student, and they may also contribute to improved retention of concepts over time, by reminding the student of previous experience and success with those concepts.

Summary

The English Language Arts (ELA) classroom can benefit from well-designed adaptive software that gives students instruction and practice, where the practice exercises are automatically evaluated, so that personalized error-specific responses can be given to guide the student in making corrections, en route to improved proficiency.

Such evaluation is significantly more challenging for software aimed at helping students to improve writing skills by having them write, but the benefits of successful automated evaluation of writing can also be significant, particularly where it provides sufficient incentive for students to practice writing more frequently.

Personalized instruction in ELA software can have a positive effect, and indeed may be an essential element of any automated system that helps students learn to write well, by giving them guided practice in actual writing.

About the Author

Dan Flickinger, Ph.D., is an internationally recognized computational linguist at Stanford's Center for the Study of Language and Information. For more than 20 years, he has been developing practical computational models of language, as described in more than 70 research publications. Dan earned his Ph.D. in linguistics from Stanford University in 1987.

References

Bender, E. M., D. Flickinger, S. Oepen, and A. Walsh (2004). "Arboretum: Using a precision grammar for grammar checking in CALL," in *Proceedings of the InSTIL/ICALL Symposium 2004*, Venice, Italy.

Flickinger, D. (2000). "On building a more efficient grammar by exploiting types," in *Natural Language Engineering* 6:1, pp. 15-28.

Flickinger, D. (2011). "Accuracy vs. Robustness in Grammar Engineering," in E.M. Bender and J.E. Arnold, eds. *Language from a Cognitive Perspective: Grammar, Usage, and Processing*, pp. 31-50. CSLI Publications, Stanford.

Suppes, P., Liang, T., Macken, E. E., and D. Flickinger (2014). "Positive technological and negative pre-test-score effects in a four-year assessment of low socioeconomic status K-8 student learning in computer-based Math and Language Arts courses," in *Computers & Education* 71.0, pp. 23-32.