The Mobile Fact and Concept Textbook System (MoFaCTS)

Philip I. Pavlik Jr., Andrew M. Olney, Amanda Banker, Luke Eglington, and Jeffrey Yarbro

Institute for Intelligent Systems, University of Memphis, Memphis, TN 38152, USA
ppavlik@memphis.edu, aolney@memphis.edu, ambanker@southwest.tn.edu, lgglngtn@memphis.edu, jyarbro2@memphis.edu

Abstract. An intelligent textbook may be considered to be an interaction layer that lies between the text and the student, helping the student to master the content in the text. The Mobile Fact and Concept Training System (MoFaCTS) is an adaptive instructional system for simple content that has been developed into an interaction layer to mediate textbook instruction and so is being transformed into the Mobile Fact and Concept Textbook System (MoFaCTS). In this project, MoFaCTS is being completely retooled to accept texts from a textbook and to automatically create cloze sentence practice content to help the student learn the material in the text. Additional features in the prototype stage include automatically generated refutational feedback for incorrect cloze responses and a dialog system, which will trigger a short conversation by a tutor to correct conceptual misunderstandings. MoFaCTS administers this content via a web browser, providing the teacher with score reports and class management tools. Because the "optimal practice" module is interchangeable and the cloze content can come from any text, the system is highly configurable for different grade levels, populations, and academic subjects. To foster faster research progress, data export supports the DataShop transaction format, which allows quick analysis of data using the DataShop tools.

Keywords: intelligent tutoring systems, e-learning, instructional design, cloze, reading comprehension, natural language processing

1 Introduction

MoFaCTS was based on the FaCT system, which was created to make faster progress on laboratory research and its translation to the classroom [7]. MoFaCTS extends the FaCT system with new features in addition to running in HTML5, which allows participants on mobile devices to use MoFaCTS. The framework of MoFaCTS is based on an implicit theory of "chunk" learning [8], which assumes that learning of chunks occurs through discrete "trials" (e.g., a single step problem or fill-in-in the blank sentence). As such, MoFaCTS departs from the tradition of model tracing tutors [9], which

focus on multistep problems of greater complexity, where the student is learning a sequence of rule applications. The simplified chunk-based approach in MoFaCTS allows the system to focus more easily on the problem selection aspect of tutoring, and how the selected sequence can be improved. In the terminology of VahLehn [10], MoFaCTS implements an outer loop of problem selection, which is being extended with inner loop functionality discussed in later sections. With regard to problem selection, MoFaCTS has been designed without strong assumptions about the optimal practice schedule. This absence of assumptions makes it easy to adapt to the needs of specific projects.

1.1 New Vision

The new vision for MoFaCTS as a textbook system for assisting in the process of learning written materials of all sorts comes from the realization that effective content generation is just as important as scheduling practice for the student. This new vision brings together the research of Pavlik, e.g., [1, 2] on optimal practice scheduling and Olney on text analysis and computational linguistics [3] and applies this to the problem of long-term learning of facts and concepts from texts. In this collaboration, we are working on a three-year IES development project to create content and practice for Hole's Human Anatomy and Physiology textbook [4] and test this practice in a community college context in the United States. One of the goals of this grant is to create a system that is content free and can mediate textbook instruction in any domain. This report on our progress highlights the systems current functionality, improvements, and the progress we are making with data collection in the classroom

2 Anatomy and Physiology

Anatomy and Physiology (AP) relies heavily on a vocabulary of mostly Latin derived words. This requires students to learn a new language while also trying to comprehend basic facts about the body and its function, an incredibly challenging task. Commonly used AP textbooks have electronic versions with some amount of interactivity to improve the student experience. The ability to search terms, highlight, make notes, etc. are often available. McGraw Hill, Wiley, and Person publishing all have online resources that provide some form of practice based on the textbook for AP courses.

The current options do not provide the flexibility available through cloze (i.e., fill in the blanks questions) using MoFaCTS. MoFaCTS can read a body of text and produce practice questions, offering an instructor many options for allowing them to tailor the text to their students' needs. By creating practice questions from any text, it allows instructors to use supplementary texts for practice as well. This allows instructors to quickly improve the breadth of resources available to educate students in an AP class.

The concepts of anatomy and physiology are foundational to many programs of study in the Health Sciences. AP courses are core requirements of Allied Health programs such as Nursing (RN) as such AP courses have high enrollment and a broad impact. Students consistently find these courses particularly challenging, and there is a low success rate. According to internal records from 2016-2018 at Southwest Tennessee Community College (Southwest), approximately 37% of API students do not earn a "C" or above, which is the grade required to apply for most Allied Health programs.

A shortage of nurses, along with a growth in job opportunities in the coming years, has been predicted [5, 6]. This places even more importance on improvement in comprehension and retention of AP material leading to greater student success. This makes better methods to educate students in this material particularly valuable.

3 Cloze Practice Creation

Content creation is a challenge for any adaptive learning system. For intelligent tutoring systems (ITS), in particular, the content creation problem has led to the development of a subfield of authoring tools by which ITS can be more efficiently created [11, 12], including tools that create content quasi-automatically by inferring it from correct and incorrect user actions [13, 14].

Text-based cloze item practice is aligned with several theoretical constructs underlying reading comprehension. The first construct is prior knowledge [15]. Prior knowledge has been shown to have significant positive effects on reading comprehension over the past 50 years [16-18]. High prior knowledge compensates for low reading skill during reading comprehension [19], and moderates the interaction between reading skill and text difficulty [20]. Text-based cloze practice enhances prior knowledge by strengthening memory for the text. The second construct is vocabulary knowledge. It is well known that specialized content areas, like AP, have their own vocabulary. New vocabulary can significantly impede comprehension even when at the relatively low levels of one new word per 20 words encountered [21]. When texts exceed this threshold, the standard implicit learning mechanisms for learning new vocabulary (i.e., guessing meaning from context) have limited effectiveness [22, 23]. Cloze instruction has similarities to both traditional flashcard instruction that pairs a word with its definition as well as richer vocabulary instruction that focuses on the contextual usage of a word, and cloze, definition instruction, and rich instruction have all been found to have positive effects on reading comprehension [24, 25]. Finally, the ICAP Hypothesis [26] predicts that overt learning activities are predictive of learning outcomes, such that interactive > constructive > active > passive activities, because of the cognitive processes that are necessarily engaged during each of these activity types. Cloze practice is by nature constructive because it requires the student to retrieve the correct word or phrase to complete the sentence.

Cloze creation from textbooks is distinct from ITS content generation in the respect that cloze creation has a strong random baseline. In other words, one can create valid cloze items simply by randomly selecting sentences from a textbook and the words to delete from those sentences (cloze targets), but there is no analogous random generation procedure for ITS. However, to make cloze items that are actually effective at promoting learning and reading comprehension [3], our approach leverages the same techniques used in our parallel work on quasi-automatic ITS generation [14]. As further discussed below, our current work is beginning to bridge the gap between MoFaCTS and ITS by adding inner loop functionality.

While there are objectively two steps of cloze creation, sentence selection, and word selection, these steps can take place in different orders depending on pedagogical goals. When the goal is vocabulary learning, word selection naturally precedes sentence se-

lection so that sentences are selected based on whether they contain the target vocabulary. In contrast, when the goal is reading comprehension, sentence selection naturally precedes vocabulary selection so that sentences are selected based on their contribution to a situation model of the text (i.e., a coherent mental model of the text; for an introduction see [15]).

Since both vocabulary learning and reading comprehension are pedagogical goals in the MoFaCTS intelligent textbook system, words and sentences are initially selected jointly. Joint selection is achieved using co-references in the text, i.e., nominal phrases that refer to the same entity. Using standard terminology in NLP, a sequence of co-referring nominal phrases is a co-reference chain. Entities that appear multiple times in the text, or equivalently, in longer co-reference chains, are more likely to be important in the text, and relations between those entities are more likely to be important in a situation model of the text. Cloze items created by using co-reference chains to select the corresponding words and sentences involved in those chains are, therefore, more likely to target the sentences and vocabulary important to a situation model of the text. Our initial selection procedure uses the length of co-reference chains and the number of chains in a given sentence to select sentences and simultaneously select the nominal phrases in these sentences as cloze targets. Sentences that contain (i.e., intersect) at least three co-reference chains of at least length two are prioritized, as a heuristic, but the combined length of chains in a sentence is otherwise used to rank sentences such that those with the summed chain length are prioritized. This approach is slightly different from our previous work that used only the heuristic with an additional heuristic based on discourse parse nuclearity [3]. The precise number of sentences and corresponding cloze targets returned are free parameters determined by the user, but if no parameters are given, defaults to approximately 50% of the sentences in the text with an average of two targets per sentence. Therefore, the initial cloze item creation process can be viewed as a type of extractive summarization [27].

Jointly selected sentences and words via co-reference chains are used to create the initial set of cloze items, but additional cloze items are created from the sentences using syntactic and semantic annotations to define additional cloze targets. These additional items can be viewed as elaborations of the discourse-driven backbone of the text defined by the initial set of coreference-generated items. Syntactic annotations in the form of dependency parses [28] are used to select the subject and objects of verbs (both direct and indirect objects) as well as the objects of prepositions. Semantic annotations in the form of semantic role labels are further used to create cloze targets that correspond to arguments of the verbal predicate [29]. The arguments are determined both by the predicate itself (e.g., sleep is intransitive) as well as the specific use of the predicate. Additionally, adjunct arguments are targeted to capture important relationships like negation, cause, direction, and time. For example, "John slept *because* he was tired," contains a causal adjunct beginning with because that would be targeted for cloze. As with syntactic annotations, the specific targeted words for cloze are nominal only, and so in this last example would not include the word *because* but rather nominal constitutents of the adjunct.

The focus on nominal entities is consistent across all cloze generation methods and reflects an emphasis on the relational properties between entities, or equivalently the propositional structure of the text. In the reading comprehension literature, this is referred to as a propositional textbase model [15], containing only the information that

was explicitly in the text. The textbase is a necessary precursor of the situation model, which is an elaboration of the textbase with information from outside the text itself, e.g., connections to prior knowledge. One way of thinking of the situation model is that it is constructed through sense-making of the text, i.e., as an explanation for how ideas in the text are related to each other and the outside world. Our current and ongoing work in cloze generation and optimal sequencing reinforces sense-making processes in order to promote the construction of a situation model for optimal reading comprehension.

Using co-reference chains to select sentences ensures that all sentences are connected by an entity, or in the case of our priority heuristic above, three different entities. The strong entity connections across sentences create a context by which MoFaCTS can sequence entity-related items in novel ways. Consider the sentences in Table 2 involving the word *gene*. As a skilled reader reads the text, they are able to integrate information about gene across sentences, even when there may be multiple intervening sentences that don't include the word *gene*. The cloze item creation approach above ensures that important words like *gene* are represented across multiple cloze items. The sequencing behavior of MoFaCTS further creates sequences of these related items that bring them closer together when a student incorrectly answers an item with the target *gene*. By bringing them closer together in the practice sequence, MoFaCTS compensates for the skill level of readers who were not previously able to make connections between separated sentences. In our current work, we are exploring additional section-based clustering in addition to this more global clustering, which we believe will support practice on smaller interconnected clusters.

Table 2. The first four sentences containing the word gene in Shier et al. (2019), chapter 24, listed with their position in the text

Position	Sentence
4	The unit of genetic information is a *gene*, which encodes a protein.
25	Messenger RNA molecules can represent different parts of a given *gene*, so that the 20,325 *genes* actually encode 100,000 to 200,000 different proteins.
35	However, information from human genome sequences and about which *genes* are expressed under specific circumstances is providing a new view of physiology as a complex interplay of *gene* functions.
40	For example, development of cardiovascular disease may reflect not only inheritance of specific gene variants that control blood pressure, blood clotting, and lipid metabolism, but also lifestyle influences such as stress, smoking, poor diet, and lack of physical exercise that may affect the expression of those genes in negative ways.

Another way in which MoFaCTS could support situation model construction is by changing the ordering of these items with respect to the text, e.g., reverse ordering. Novel orderings both build a more robust memory representation for the items that is independent of the text order and allow the reader to engage in sense-making beyond that licensed by the original order of the text. One way of conceptualizing this property is that while knowledge can be viewed as structured in a graph, a text is inherently linear in how it communicates that graph, i.e., a text represents just a single possible

traversal, or walk, of the graph. The connectedness of our cloze items ensures that MoFaCTS can take many different traversals of the same graph, allowing the student to make connections between concepts that were not foregrounded in the original presentation.

Finally, we have added several feedback features that are bridging the gap between MoFaCTS as an outer loop system and inner-loop systems like ITS. The default MoFaCTS feedback is correct/incorrect, where incorrect is further supplemented by the correct answer. Student answers need not be exact: edit distance is used to give students credit for "close" answers, and acronym mapping gives students credit for using an acronym in place of a phrase or vice versa. However, the default MoFaCTS feedback does not analyze the student's error and provide error-specific feedback.

We have implemented two forms of elaborated feedback, refutational feedback, and tutorial dialogue. Refutational feedback currently uses glossary-driven natural language generation to provide a paragraph response to a student's incorrect answer that defines both the nominal phrase in the student's incorrect answer (concept B) and the nominal phrase in the correct answer (concept A). For example, if the following cloze item were presented, "The brain connects to the spinal cord through the *brain stem*.", and the student replied with *nervous system*, then the system would reply with, "Nervous system is not right. The right answer is brain stem. The difference is that the nervous system is a network of cells that sense and respond to stimuli in ways that maintain homeostasis, and the brain stem is a portion of the brain that includes the midbrain, pons, and medulla oblongata." The rationale behind this approach is that when a student gives an incorrect answer, they are potentially revealing three knowledge deficits: concept A, concept B, and the difference between A and B. The refutational feedback addresses all three potential deficits, whereas the default feedback only addresses concept A. Addressing all three potential knowledge deficits simultaneously may increase the efficiency of MoFaCTS by remediating erroneous knowledge before it is explicit. In the near term, we plan to diagnose errors and construct feedback at a finer resolution using concept maps [30].

Tutorial dialogues, launching in response to a student error, take the feedback of MoFaCTS even closer to an inner loop by providing step-level instruction. Using question generation techniques [14], tutor hints and prompts are generated for each cloze item, and then the questions are delivered according to the AutoTutor-type sequence of hint, prompt, and elaboration [31]. For example, the cloze item, "Connections from the *cerebral cortex,* including the limbic system, can influence autonomic centers and increase both sympathetic and parasympathetic activities.", would be converted into the hint "Tell me about autonomic centers." and prompt "What can influence autonomic centers and increase both sympathetic and parasympathetic activities?", followed by the elaboration, "It is important to remember that connections from the cerebral cortex, including the limbic system, can influence autonomic centers and increase both sympathetic and parasympathetic activities." When the student types in their answer to each hint or prompt, it is assessed for correctness using textual entailment [32]. This textual entailment approach gives two ratings, entailment and contradiction, which are used to determine the polarity of feedback (positive vs. negative), the magnitude of the feedback (e.g., positive, positive-neutral, or neutral), and whether the student has answered correctly enough that the dialogue should terminate and normal MoFaCTS operation should continue. As illustrated by this example, the current tutorial dialogue feedback

remediates cloze item errors but does not create a refutation. Our ongoing and future work is investigating how to create refutational dialogues.

3.1 Syllabification and Hints

Hints in MoFaCTS are given by syllabifying the cloze answer and then displaying one or more of the syllables as a hint to the user. Syllables were chosen as the hint methodology due to them offering a consistent, meaningful unit that helps strengthen the student's phonetic and orthographic representations of words. By enriching these representations, along with the semantic representations targeted by cloze items, the overall quality of a cloze word's mental representation should increase as detailed by the Lexical Quality Hypothesis [33]. This could be especially important in scientific fields such as Anatomy and Physiology, where words commonly share meaningful affixes (e.g. 'neurocyte' and 'cytoplasm'). For detailed information on how syllabification is performed, see Ash [34]. An example is shown in Figure 1.

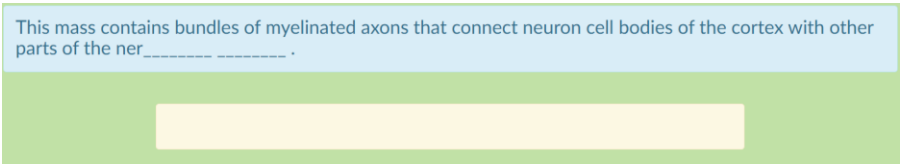

Figure 1. Example of the syllable "ner" from "nervous system" displayed to the user as a hint.

The strength of these hints (or "cues") have important consequences on learning. Providing strong hints may increase the likelihood that a student responds correctly and quickly and may allow them to answer more questions. Providing strong hints can be especially important when students are new to an educational topic. However, providing overly strong hints may make the task too easy and reduce the learning for that trial [35]. On the other hand, a very weak cue may promote more learning, but also introduce a higher risk of failure (and thus increased time cost). Balancing the relative gains and costs associated with hint difficulty is important, and there is some evidence that adaptively cueing can benefit memory. For instance, Fiechter & Benjamin [36] provided evidence that varying the strength of cues improved student learning of English-Iñupiaq word pairs (e.g., tea–saiyu). They found that increasing hint difficulty as practice progressed (e.g., tea-sai__ vs. tea-s___) provided better learning gains than fixing the difficulty of the hints for all trials.

Results of the different hint conditions (Spring 2020 data) are shown in Figure 2. Since the selection of hint condition for each item (only those responses with 3 or more syllables were eligible) was fully random, and 24 students did more than 100 trials (12,407 total trials aggregated), the data was well configured for a repeated measures comparison of the causal effect of the number of syllables on performance. Using repeated measures ANOVA, we compared the subject means for first 2 trials in each of the 3 conditions and found a highly significant difference (Wilk's Lambda = .309, $F(2,22) = 24.596$, $p < .0005$, $\eta_p^2 = .691$). Post hoc comparisons (Bonferroni corrected) showed no hint was significantly different from 1 or 2 hints ($p < .0005$), and 1 and 2 hint syllable conditions were also significantly different from each other ($p = .011$).

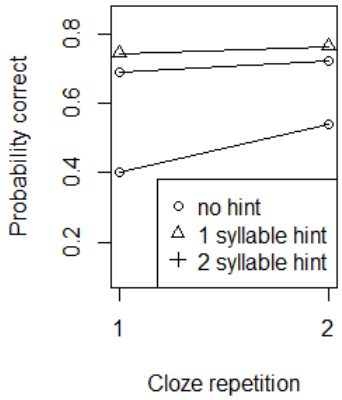

Figure 2. Performance differences by condition (Spring2020), where each cloze for each subject randomly received either 0, 1, or 2 syllables hint for both of the first 2 trials.

The model implemented for Summer 2020 makes use of these results by computing expected recall probability as a function of hinting. This adaptivity is expected to improve the efficiency of practice as well as increase the motivation of the student (under the assumption that successes with more moderate difficulty items are preferred relative to greater difficulty).

We intend to further quantify hint strength by adding 3 additional factors to the model: Pointwise Mutual Information (PMI), syllable length, and syllable position. These are described in detail below. PMI will be used to measure the mean association between syllables within a cloze answer. The idea is that the higher the association between a syllable and the other syllables within an answer, the stronger the hint it provides. To perform the mean PMI calculation, we first create cloze items and syllabify each answer within the clozes. This data is then used to calculate the probabilities of occurrences and co-occurrences for each syllable. Considering syllable length means when a syllable contains a greater proportion of a word's characters, it contains more information about that word. This makes it such that a longer syllable should naturally serve as a stronger hint than a shorter one.

4 Optimized Delivery

Determining the order in which to present items is an active area of research [37, 38]. There are general recommendations [39], as well as more specific model-based approaches that incorporate theories of spacing, testing, and forgetting [1, 40]. Model-based approaches that adapt according to prior student performance outperform more fixed schedules [1]. MoFaCTS offers the ability to schedule practice according to specific difficulty thresholds. For instance, in MoFaCTS one could practice a set of items with the order determined by an optimal efficiency threshold (OET) of .9. This would mean that whatever item had an estimated recall closest to .9 would be practiced next. Distance from .9 could be absolute distance or ceiling (e.g., whatever is closest to, but less than). This threshold effect is shown in Figure 3, which shows performance across

repetitions in our Fall data. Despite this being a memory task that could become very easy with repetitions, the algorithm is able to maintain practice at a relatively constant difficulty level.

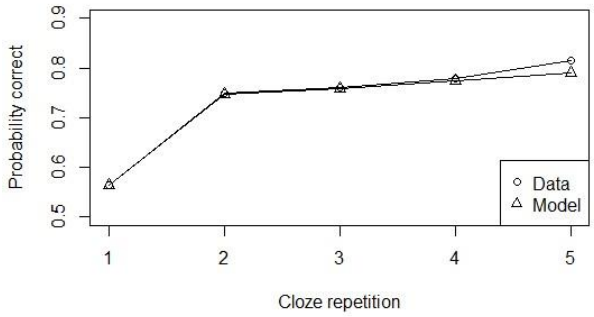

Figure 3. Practice performance across repetitions of an item. OET was set to .7. Model is the post hoc fit used to derive the Summer 2020 practice model parameters.

Of course, recall probability needs to be known to implement such scheduling. MoFaCTS allows custom models to be used to estimate recall probabilities of all practice items and updates those estimates on every trial. Models can be as simple or complicated as desired. Model choice has substantial influence over the efficacy of the threshold chosen. For instance, if practicing whatever item is closest to a .9 threshold, the two models described in Equations 1 and 2 below would lead to very different behavior. Equation 2 includes a forgetting parameter, and so even if an item was practiced to above .9 in one session, that item would eventually fall below .9 again and be practiced. In contrast, Equation 1 simply uses a count of attempts and a slope to estimate knowledge. Memory decay is not assumed in Equation 1, and thus items can actually be entirely dropped from practice. Together, custom models and difficulty thresholds allow substantial flexibility to apply known models, e.g. [37, 41] to improve learning, but also to carry out experimentation [42]. There are broad implications to learning with MoFaCTS guided by a learner model and a practice difficulty threshold. For instance, if the model includes parameters in which recall probability is influenced by time (e.g., spacing, decay), then different practice thresholds will induce different (adaptive) spacing intervals.

$$\beta_i N_{ij} \tag{1}$$

$$\beta_i N_{ij} t_{ij}^{-d} \tag{2}$$

Equation 1 includes a learning parameter β for each KC *i*, multiplied by the number of prior attempts *N* for KC *i* made by student *j*. In Equation 2, this value is multiplied by the elapsed time *t* since the KC was first practiced by the student *j*, with a decay parameter *d*.

Our current model for the AP project has three main assumptions that allow us to infer the OET accurately. First, the model has an assumption of a quadratic effect of prior practices as a function of their difficulty [42, 43], using different effect curves for success and failure, which makes the effect of successes and failure different quantities, like the performance factors analysis model [44], second, there is the assumption of forgetting as a function of recency, and third, there is the assumption of differential time costs for failures and success. Using the model of these effects, we inferred the optimum values. Our OET was found to be .7 in Fall 2019, .71 in Spring 2020, and .72 for Summer 2020, showing a remarkable consistency despite these inferences differing greatly in the input data and the model used. Figure 4 illustrates the OET curve in use for Summer 2020. The input model was calibrated with Fall 2019 and Spring 2020 data (N = 26,225).

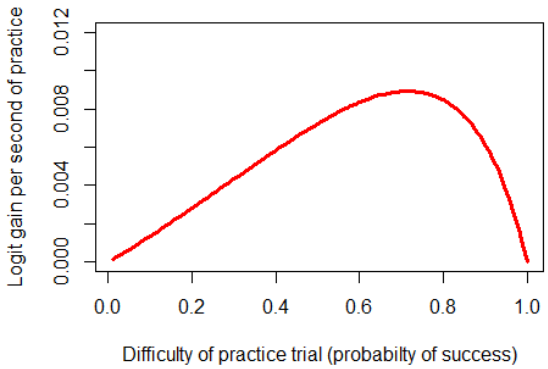

Figure 4. Cloze practice efficiency estimated from the model of a Fall 2019 and Spring 2020 community college sample studying AP. The maximum is at .72.

5 MoFaCTS Instructor Management Functions

Most instructor functionality is new to MoFaCTS and includes the ability to create content, create classes, assign content to classes, and lookup student performance in classes for assigned content units. We will be working to improve these functions over the duration of the grant. One of the main challenges in this is to provide easy access for both students and teachers. Our current system is configured to allow student to use their university single sign on identity to log into our system.

6 MoFaCTS Content Management Functions

MoFaCTS has two primary unit types, learning (described in the Optimized Delivery section above) and assessment, which define its two main modes of application. Both kinds of units are specified in the control file for each "tutor," which is called the tutor definition file (TDF). Each tutor definition file begins with a number of preliminaries,

including the initial randomization commands. To enable comparisons of different assessment or learning conditions, the system also automatically randomizes into any number of between-subjects conditions. This choice is recorded in the data for each subject and reinstated when they begin new sessions from the same root TDF, so multi-session between-subjects comparisons with counter-balancing are easily enabled.

The assessment unit allows traditional experimental designs and requirements (e.g., counter-balancing) to be enforced. Assessment units allow for complex schedules of content, where the TDF author has specified the number of repetitions and the location in the sequence for each repetition of each item. Each repetition may be a test with or without feedback or a passive study opportunity. Assessment units may be used for quizzes in a classroom setting or for experiments looking at practice, forgetting, learning, and/or recall. In the case of AP, this type of unit is used for class surveys throughout the development process. In an experimental context, the system allows additional sequence level randomization, to make sure blocks of the same items are individually randomized so that spacing conditions are not predictable. Any number of assessment units can be strung together, which allows pretest, practice, and posttest portions to be organized individually to compose a larger experiment.

6.1 Item Types

The system supports two other main forms of test items in addition to cloze: the multiple-choice items (which appear in button form for touchscreen responsiveness) and the short-answer items. Basic feedback for all item types displays the correct answer for a fixed period of time or until the user hits the spacebar, as specified in the TDF. If the trial is a short answer item, more complex branching feedback is allowed, which compares the response with a number of wrong responses, each of which has specific feedback text in the stimulus file.

Since both the system and the user may be frustrated and deterred in their goals by incorrectly marked cloze or short answer responses, the system provides a few ways to identify correct responses with some flaws or ambiguity. These include partial matching using regular expressions, simple Levenshtein proportion errors, or Levenshtein proportion for multiple synonyms. Each of these methods offers different advantages depending on the test type. Regular expressions allow answer specification to pick up the presence of keywords for short answer responses, to automatically score relatively complex responses. Levenshtein proportion marks an item correct if some proportion of the letters are correct (e.g., 75%).

6.2 Datashop Export and Amazon Turk Integration

The system provides native export to the DataShop tab-delimited format style with several custom fields. This functionality means that data collected in the system can be immediately imported into DataShop for analysis, storage, and/or presentation [45]. As part of the new LearnSphere project, the DataShop is being expanded to include a graphical workflow analysis tool with multiple methods (http://learnsphere.org/). MoFaCTS users will be able to take advantage of these resources immediately. Further, there is a library of prior analyses already shared within the community for DataShop formatted files (https://pslcdatashop.web.cmu.edu/ExternalTools).

The system also provides integration with Amazon's Mechanical Turk (MTurk) service. This integration was added to ease the administrative burden often encountered when running experiments with large numbers of participants recruited via MTurk. A researcher can oversee the experiment via a management screen within MoFaCTS that shows the current progress of all participants. From the same screen, the researcher may approve payment for a participant's work and/or pay a post-payment bonus. If using the "lockout conditions" discussed previously, researchers may craft an automated message that the system will send to Mechanical Turk users when their lockout expires (e.g., email a reminder after a one-week retention interval).

6.3 Client/ Server Architecture

MoFaCTS was built using Meteor, a framework based on Node.js, which uses a single programming language (JavaScript) for both the client and server logic. Communication between the two sides of the architecture is handled transparently by the framework. This architecture conveniently off-loads any complex computations needed to compute practice schedules to the client machine, which allows much larger numbers of users to interact with the system simultaneously.

7 Conclusions

MoFaCTS was created as a research tool to investigate the effect of instructional sequence manipulations. The system is released on bitbucket.org as open-source software (https://bitbucket.org/ppavlik/MoFaCTS/overview). As development to create a textbook instruction system continues, we welcome collaborators interested in new domains and contexts. Continued development is focused on refining existing capabilities and completing the addition of the refutation and dialogue features for remediating student misconceptions.

8 Acknowledgments

This work was supported by the Institute of Education Sciences (IES; R305A190448). Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of IES. This work was also supported by the National Science Foundation (NSF) Data Infrastructure Building Blocks project, Learner Data Institute project, and DataWhys project, under Grant Nos. (1443068, 1934745, and 1918751) and the University of Memphis Institute for Intelligent Systems.

9 References

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

2. Pavlik Jr., P.I., Kelly, C., Maass, J.K.: Using the Mobile Fact and Concept Training System (Mofacts). In: Micarelli, A., Stamper, J. (eds.): Proceedings of the 13th International Conference on Intelligent Tutoring Systems, 247-253. Springer, Switzerland (2016)
3. Olney, A.M., Pavlik, P.I., Maass, J.K.: Improving Reading Comprehension with Automatically Generated Cloze Item Practice. In: André, E., Baker, R., Hu, X., Rodrigo, M.M.T., du Boulay, B. (eds.): Proceedings of Artificial Intelligence in Education: 18th International Conference, 262-273. Springer International Publishing, Wuhan, China (2017)
4. Shier, D., Butler, J., Lewis, R.: Hole's Human Anatomy and Physiology. McGraw-Hill Education (2019)
5. Hogan, A., Roberts, B.: Occupational Employment Projections to 2024. In: , B.o.L.S. (ed.): Monthly Labor Review (2015), https://www.bls.gov/opub/mlr/2015/article/occupational-employment-projections-to-2024.htm
6. Juraschek, S.P., Zhang, X., Ranganathan, V., Lin, V.W.: United States Registered Nurse Workforce Report Card and Shortage Forecast. American Journal of Medical Quality: 27, 241-249. (2012)
7. Pavlik Jr., P.I., Presson, N., Dozzi, G., Wu, S.-m., MacWhinney, B., Koedinger, K.R.: The Fact (Fact and Concept Training) System: A New Tool Linking Cognitive Science with Educators. In: McNamara, D., Trafton, G. (eds.): Proceedings of the Twenty-Ninth Annual Conference of the Cognitive Science Society, 1379–1384. Lawrence Erlbaum, Mahwah, NJ (2007)
8. Johnson, N.F.: The Role of Chunking and Organization in the Process of Recall. The psychology of learning and motivation: 4, 171-247. (1970)
9. Anderson, J.R., Pelletier, R.: A Development System for Model-Tracing Tutors. Proceedings of the International Conference of the Learning Sciences, 1-8. Evanston, IL (1991)
10. VanLehn, K.: The Behavior of Tutoring Systems. International Journal of Artificial Intelligence in Education: 16, 227-265. (2006)
11. Dermeval, D., Paiva, R., Bittencourt, I.I., Vassileva, J., Borges, D.: Authoring Tools for Designing Intelligent Tutoring Systems: A Systematic Review of the Literature. International Journal of Artificial Intelligence in Education: 28, 336-384. (2018)
12. Murray, T.: An Overview of Intelligent Tutoring System Authoring Tools: Updated Analysis of the State of the Art. In: Murray, T., Blessing, S.B., Ainsworth, S. (eds.): Authoring Tools for Advanced Technology Learning Environments: Toward Cost-Effective Adaptive, Interactive and Intelligent Educational Software, 491-544. Springer Netherlands, Dordrecht (2003)
13. Matsuda, N., Cohen, W.W., Koedinger, K.R.: Teaching the Teacher: Tutoring Simstudent Leads to More Effective Cognitive Tutor Authoring. International Journal of Artificial Intelligence in Education: 25, 1-34. (2015)
14. Olney, A.M.: Using Novices to Scale up Intelligent Tutoring Systems. Interservice/Industry Training, Simulation, and Education Conference (I/ITSEC) 2018 (2018)
15. McNamara, D.S., Magliano, J.P.: Toward a Comprehensive Model of Comprehension. 297-384. Academic Press, New York (2009)
16. Lipson, M.Y.: Learning New Information from Text: The Role of Prior Knowledge and Reading Ability. Journal of Reading Behavior: 14, 243-261. (1982)
17. Ahmed, Y., Francis, D.J., York, M., Fletcher, J.M., Barnes, M., Kulesz, P.: Validation of the Direct and Inferential Mediation (Dime) Model of Reading Comprehension in Grades 7 through 12. Contemporary Educational Psychology: 44-45, 68 - 82. (2016)
18. Bransford, J.D., Johnson, M.K.: Contextual Prerequisites for Understanding: Some Investigations of Comprehension and Recall. Journal of Verbal Learning and Verbal Behavior: 11, 717-726. (1972)

19. Recht, D.R., Leslie, L.: Effect of Prior Knowledge on Good and Poor Readers' Memory of Text. Journal of Educational Psychology: 80, 16-20. (1988)
20. Ozuru, Y., Dempsey, K., McNamara, D.S.: Prior Knowledge, Reading Skill, and Text Cohesion in the Comprehension of Science Texts. Learning and Instruction: 19, 228-242. (2009)
21. Laufer, B.: Lexical Thresholds for Reading Comprehension: What They Are and How They Can Be Used for Teaching Purposes. TESOL Quarterly: 47, 867-872. (2013)
22. Fang, Z.: The Language Demands of Science Reading in Middle School. International Journal of Science Education: 28, 491-520. (2006)
23. Nagy, W., Townsend, D.: Words as Tools: Learning Academic Vocabulary as Language Acquisition. Reading Research Quarterly: 47, 91-108. (2012)
24. National Reading Panel: Report of the National Reading Panel. Teaching Children to Read: An Evidence-Based Assessment of the Scientific Research Literature on Reading and Its Implications for Reading Instruction (00-4769). National Institute of Child Health & Human Development, Washington, DC (2000)
25. McKeown, M.G., Beck, I.L., Omanson, R.C., Pople, M.T.: Some Effects of the Nature and Frequency of Vocabulary Instruction on the Knowledge and Use of Words. Reading Research Quarterly: 20, 522-535. (1985)
26. Chi, M.T.H., Wylie, R.: The Icap Framework: Linking Cognitive Engagement to Active Learning Outcomes. Educational Psychologist: 49, 219-243. (2014)
27. Nenkova, A., McKeown, K.: Automatic Summarization. Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Tutorial Abstracts, 3. Association for Computational Linguistics, Portland, Oregon (2011)
28. De Marneffe, M.-C., Manning, C.D.: Stanford Typed Dependencies Manual. Technical report, Stanford University (2008)
29. Palmer, M., Gildea, D., Kingsbury, P.: The Proposition Bank: An Annotated Corpus of Semantic Roles. Computational linguistics: 31, 71-106. (2005)
30. Olney, A.M., Graesser, A.C., Person, N.K.: Question Generation from Concept Maps. Dialogue & Discourse: 3, 75-99. (2012)
31. Nye, B.D., Graesser, A.C., Hu, X.: Autotutor and Family: A Review of 17 Years of Natural Language Tutoring. International Journal of Artificial Intelligence in Education: 24, 427-469. (2014)
32. Gardner, M., Grus, J., Neumann, M., Tafjord, O., Dasigi, P., Liu, N.F., Peters, M., Schmitz, M., Zettlemoyer, L.: Allennlp: A Deep Semantic Natural Language Processing Platform. ACL 2018, 1. (2018)
33. Perfetti, C.A., Hart, L.: The Lexical Quality Hypothesis. Precursors of functional literacy: 11, 67-86. (2002)
34. Ash, S., Lin, D.: Grapheme to Phoneme Translation Using Conditional Random Fields with Re-Ranking. International Conference on Text, Speech, and Dialogue, 314-325. Springer (2016)
35. Carpenter, S.: Cue Strength as a Moderator of the Testing Effect: The Benefits of Elaborative Retrieval. Journal of experimental psychology. Learning, memory, and cognition: 35, 1563-1569. (2009)
36. Fiechter, J.L., Benjamin, A.S.: Techniques for Scaffolding Retrieval Practice: The Costs and Benefits of Adaptive Versus Diminishing Cues. Psychonomic Bulletin & Review: 26, 1666-1674. (2019)

37. Walsh, M.M., Gluck, K.A., Gunzelmann, G., Jastrzembski, T., Krusmark, M., Myung, J.I., Pitt, M.A., Zhou, R.: Mechanisms Underlying the Spacing Effect in Learning: A Comparison of Three Computational Models. Journal of Experimental Psychology: General: 147, 1325-1348. (2018)
38. Ridgeway, K., Mozer, M.C., Bowles, A., Stone, R.: Forgetting of Foreign-Language Skills: A Corpus-Based Analysis of Online Tutoring Software. Cognitive Science Journal.(Accepted for publication). (2016)
39. Miyatsu, T., Nguyen, K., McDaniel, M.A.: Five Popular Study Strategies: Their Pitfalls and Optimal Implementations. Perspectives on Psychological Science: 13, 390-407. (2018)
40. Lindsey, R.V., Shroyer, J.D., Pashler, H., Mozer, M.C.: Improving Students' Long-Term Knowledge Retention through Personalized Review. Psychological Science. (2014)
41. Pavlik Jr, P.I., Eglington, L.G., Harrell-Williams, L.M.: Generalized Knowledge Tracing: A Constrained Framework for Learner Modeling. Manuscript submitted for publication (https://arxiv.org/abs/2005.00869). (preprint)
42. Cao, M., Pavlik Jr, P.I., Bidelman, G.M.: Incorporating Prior Practice Difficulty into Performance Factor Analysis to Model Mandarin Tone Learning. In: Lynch, C., Merceron, A., Desmarais, M., Nkambou, R. (eds.): Proceedings of the 11th International Conference on Educational Data Mining, 516-519. (2019)
43. Cao, M., Pavlik Jr, P.I.: Using a Variant of the Performance Factors Analysis Model for Adaptive Training on Mandarin Tones. In: Xiangen, H., al., E. (eds.): Third International Conference on Artificial Intelligence and Adaptive Education 2019, Beijing, China (2019)
44. Pavlik Jr., P.I., Cen, H., Koedinger, K.R.: Performance Factors Analysis -- a New Alternative to Knowledge Tracing. In: Dimitrova, V., Mizoguchi, R., Boulay, B.d., Graesser, A. (eds.): Proceedings of the 14th International Conference on Artificial Intelligence in Education, 531–538. Brighton, England (2009)
45. Koedinger, K.R., Baker, R.S., Cunningham, K., Skogsholm, A., Leber, B., Stamper, J.: A Data Repository for the Edm Community: The Pslc Datashop. In: Romero , C., Ventura, S., Pechenizkiy, M. (eds.): Handbook of Educational Data Mining, Vol. 43. CRC Press, Boca Raton (2010)