APPENDIX A

The probabilistic graphical model in Figure 1 was carefully crafted so that Bayesian inference can be conducted analytically. In this appendix, we first show that

$$\begin{aligned} & P(w_1, \ldots, w_n \mid y_{1:t}, a_{1q_1}, \ldots, a_{tq_t}) \\ = & \prod_j P(w_j \mid y_{1:t}, a_{1q_1}, \ldots, a_{tq_t}) \end{aligned}$$

In other words, we show that W_j is conditionally independent of $W_{j'}$ given $Y_{1:t}$ and $A_{1q_1}, \ldots, A_{tq_t}$ for all $j' \neq j$. To prove this conditional independence we will show that every undirected path (see the graph in Figure 1) between W_j and $W_{j'}$ is d-separated. (For brevity, we will omit the word "undirected".) First, note that it suffices to show that every path without cycles betwen W_j and $W_{j'}$ is d-separated: any path with cycles can be trivially converted to a path without cycles, and if the simplified path (with no cycles) is d-separated, then so too will be the original path.

Our proof proceeds in two parts. In Part 1, we will show that any path P from W_j to $W_{j'}$ must contain two nodes $A_{\nu j}$ and $A_{\nu j'}$ for some timestep ν such that $j' \neq j$. In Part 2 we will use the fact that, at every round ν, the student observes at most one node $A_{\nu j}$ to show that the other, *unobserved* node $A_{\nu j'}$ d-separates W_j from $W_{j'}$.

Part 1: Notice that the only nodes to which W_j is connected are $A_{1j}, \ldots, A_{tj}$; hence P must start with the nodes $W_j A_{\nu j}$ for some ν. Now, consider the next node in P after $A_{\nu j}$: the only nodes to which $A_{\nu j}$ is connected are C_ν and W_j. We can ignore the latter possibility because a path that starts out as $W_j A_{\nu j} W_j$ would contain a cycle. Hence, P must start out as $W_j A_{\nu j} C_\nu$. From the graphical model it is clear that any path from C_ν to $W_{j'}$ must eventually proceed through some node $A_{\nu j'}$. The only remaining question is whether $j' = j$ or $j' \neq j$. However, we can discard the former possibility because that would result in a cycle. Thus every path P from W_j to $W_{j'}$ must contain two nodes $A_{\nu j}$ and $A_{\nu j'}$ such that $j' \neq j$ for some timestep ν.

Part 2: To finally prove d-separation between W_j and $W_{j'}$, we note that, at each timestep ν, at most one of $A_{\nu j}$ and $A_{\nu j'}$ can be observed by the student because only one query is "answered" at each timestep. Since at least one of those two nodes is unobserved, and since none of the A nodes has any descendants, then P is d-separated (by the "inverted fork" rule) by either $A_{\nu j}$ or $A_{\nu j'}$. Since this is true of any path P without cycles, we conclude that W_j is d-separated from $W_{j'}$.

APPENDIX B

Given that the meaning of each word can be inferred independently, we can now derive the equation representing a Bayesian learner's belief update: Let M_t be a matrix of random variables specifying the student's belief at time t about the words' meanings, where entry M_{tji} specifies the student's posterior belief $P(W_j = i \mid y_{1:t}, a_{1q_1}, \ldots, a_{tq_t})$ that word j means concept i given the images and answers she has observed. As shown in the previous appendix, the joint posterior distribution of the meanings of all words is equal to the product of the marginal posterior distributions. Now, consider the marginal posterior distribution $P(W_j = i \mid y_{1:t}, a_{1q_1}, \ldots, a_{tq_t})$, and suppose that at timestep t the teacher teaches a word $q_t \neq j$. Then due to the conditional independence properties of the graphical model in Figure 1,

$$\begin{aligned} M_{t+1,ji} &\doteq P(W_j = i \mid y_{1:t}, a_{1q_1}, \ldots, a_{tq_t}) \\ &= P(W_j = i \mid y_{1:t}, a_{1q_1}, \ldots, a_{t-1,q_{t-1}}) \\ &= P(W_j = i \mid y_{1:t-1}, a_{1q_1}, \ldots, a_{t-1,q_{t-1}}) \\ &\quad \text{(cond. indep. from graphical model)} \\ &= M_{tji} \end{aligned}$$

In other words, the posterior distribution of W_j is equal to the prior distribution for every timestep t when the teacher teaches a word $q_t \neq j$.

On the other hand, if the teacher teaches word $q_t = j$ at timestep t, then

$$\begin{aligned} & P(W_j = i \mid y_{1:t}, a_{1q_1}, \ldots, a_{tq_t}) \\ \propto\ & P(a_{tq_t} \mid W_j = i, y_{1:t}, a_{1q_1}, \ldots, a_{t-1,q_{t-1}}) \\ & P(W_j = i \mid y_{1:t}, a_{1q_1}, \ldots, a_{t-1,q_{t-1}}) \\ =\ & P(a_{tj} \mid W_j = i, y_{1:t}, a_{1q_1}, \ldots, a_{t-1,q_{t-1}}) \\ & P(W_j = i \mid y_{1:t}, a_{1q_1}, \ldots, a_{t-1,q_{t-1}}) \\ =\ & P(a_{tj} \mid W_j = i, y_{1:t}, a_{1q_1}, \ldots, a_{t-1,q_{t-1}}) \\ & P(W_j = i \mid y_{1:t-1}, a_{1q_1}, \ldots, a_{t-1,q_{t-1}}) \\ =\ & P(a_{tj} \mid W_j = i, y_t) P(W_j = i \mid y_{1:t-1}, a_{1q_1}, \ldots, a_{t-1,q_{t-1}}) \\ & \text{(by cond. indep. from graphical model)} \\ =\ & M_{tji} P(a_{tj} \mid W_j = i, y_t) \end{aligned}$$

To compute $P(a_{tj} \mid W_j = i, y_t)$, we handle the case that $A_{tj} = 1$ (i.e., Y_t represents word Q_t) and $A_{tj} = 0$ (i.e., Y_t does not represent word Q_t) separately. For the former case,

$$\begin{aligned} & P(A_{tj} = 1 \mid W_j = i, y_t)] \\ =\ & \sum_{i'=1}^{m} P(A_{tj} = 1 \mid C_t = i', W_j = i, y_t) \\ & P(C_t = i' \mid W_j = i, y_t) \\ =\ & \sum_{i'=1}^{m} P(A_{tj} = 1 \mid C_t = i', W_j = i) P(C_t = i' \mid y_t) \\ =\ & P(C_t = i \mid y_t) \end{aligned}$$

since $i' = i$ is the only value of i' that contributes positive probability mass to the sum. The latter case ($A_{tj} = 0$) is then simply $1 - P(C_t = i \mid y_t)$.

Combining these two cases, we get:

$$M_{t+1,ji} \propto M_{tji} P(C_t = i \mid y_t)^{a_{tj}} (1 - P(C_t = i \mid y_t))^{(1 - a_{tj})} \tag{1}$$

In other words, if the teacher teaches word j at timestep t, then the student updates her belief about the meaning of that word: if the teacher says image y_t *does* represent word j ($A_{tj} = 1$), then the student increases the probability that word j means any concept i that is shown in the image with high probability. If, on the other hand, the teacher said y_t does *not* represent word j ($A_{tj} = 0$), then the student *decreases* the probability that word j means any concept i that is shown in the image with high probability.

APPENDIX C

In this appendix we provide more details on how the macro- and micro-controllers described in Sections 7.1 and 7.2.

C.1 Macro-controller

We define x_t to consist of the following features:

- For each word j, the expected (w.r.t. the teacher's particles) "goodness" $\sum_p \omega_p g(M_{tj}^{(p)})$ of the student's belief about word j, where *goodness* is defined as

$$g(M_{tj}^{(p)}) \doteq M_{tji}^{(p)} \text{ for } i = W_j \tag{2}$$

  and $M_{tj}^{(p)}$ is the student's belief at time t about word j according to particle p. In other words, the goodness of the student's belief about the meaning of word j is the probability she assigns to the correct concept.
- The teacher's uncertainty about the student's beliefs (summed over all words), i.e., $\sum_p \omega_p \sum_j \sigma(M_{tj}^{(p)})$, where the uncertainty σ is defined as

$$\sigma(M_{tj}^{(p)}) \doteq (M_{tj}^{(p)} - \overline{M_{tj}})^\top (M_{tj}^{(p)} - \overline{M_{tj}})$$

  and where

$$\overline{M_{tj}} \doteq \sum_p \omega_p M_{tj}^{(p)}$$

  Note that the uncertainty is over the *teacher's* belief about the student's belief; it is not over the student's belief itself. (The latter uncertainty is captured by the "goodness" defined above.)
- A bias term (constant 1).

C.2 Micro-controller

We define "total uncertainty" as:

$$\sum_p \omega_p \sum_j \sigma(M_{tj}^{(p)}) + \sum_p \omega_p (\alpha_t^{(p)} - \overline{\alpha_t})^2 + \sum_p \omega_p (\beta_t^{(p)} - \overline{\beta_t})^2$$

where $\overline{\alpha_t} \doteq \sum_p \omega_p \alpha_t^{(p)}$ and $\overline{\beta_t} \doteq \sum_p \omega_p \beta_t^{(p)}$, and where $\alpha_t^{(p)}$ and $\beta_t^{(p)}$ are the student's absorption and belief update strength at time t according to particle p. This metric includes uncertainty not just over the student's belief, but also over student parameters α_t and β_t.

APPENDIX D

To enable a fair comparison between Bayesian Knowledge Tracing (BKT) and AOTAOL for predicting students' test scores (see Section 9.5), we implemented BKT in the following way: One model (consisting of parameters p, g, and s – see below) was trained for each of the $n = 10$ words, and each model consisted of a Hidden Markov Model with two latent states – "learned" and "unlearned". If the student is in the "learned" state for word j, then with probability $(1 - s)$ she/he answers a question about word j correctly on a test, where s is the *slip* probability. If the student is in the "unlearned" state, then she/he answers a question about word j correctly with probability g, which is the *guess* probability. Whenever the student is given a "teach" action about word j and she/he is currently in the "unlearned" state, then the student will transition to the "learned" state with probability p. The "learned" state is a trap state – the probability of transitioning to "unlearned" is fixed at 0. The "test" actions do not alter the student's state. Finally, in our implementation of BKT, all the "ask" actions are ignored completely – to model these actions would require a more sophisticated model than BKT (such as AOTAOL itself) that can consider how the student responds to specific queries about which image of two images more likely represents a particular word.

Training: Training was conducted using data (from 40 students) collected to estimate the time costs of each action (see Section 8). For each of the $n = 10$ words, we used all of these students' observations – consisting of whether or not (1 or 0, respectively) the student answered a test question about word j correctly on a test, or a "dummy" observation (2) for "teach" actions – to train a BKT model using maximum likelihood estimation. Hence, for each word, there was one observation per test for each student (and most students took the test several times before completing the task). To address the issue of local minima, we randomly tried 5 different starting points for each model and chose the one that maximized the data likelihood. HMM optimization was performed using Kevin Murphy's Hidden Markov Model (HMM) Toolbox for Matlab.

Test score estimation: For each student and for each test at time t, all observations for that student up through timestep $t - 1$ were used to estimate the probability that the student answers a test question about each word $j \in \{1, \ldots, n\}$ correctly. The average probability over all n words was then taken as the expected score for that test.