Machine Beats Human at Sequencing Visuals for Perceptual-Fluency Practice

Ayon Sen (asen6@wisc.edu)[1], Purav Patel[2], Martina A. Rau (marau@wisc.edu)[2],
Blake Mason[3], Robert Nowak[3], Timothy T. Rogers[4], Xiaojin Zhu[1]
[1] Department of Computer Sciences, [2] Department of Educational Psychology,
[3] Department of Electrical and Computer Engineering, [4] Department of Psychology
University of Wisconsin-Madison

ABSTRACT

In STEM domains, students are expected to acquire domain knowledge from visual representations that they may not yet be able to interpret. Such learning requires perceptual fluency: the ability to intuitively and rapidly see which concepts visuals show and to translate among multiple visuals. Instructional problems that engage students in nonverbal, implicit learning processes enhance perceptual fluency. Such processes are highly influenced by sequence effects. Thus far, we lack a principled approach for identifying a sequence of perceptual-fluency problems that promote robust learning. Here, we describe a novel educational data mining approach that uses machine learning to generate an optimal sequence of visuals for perceptual-fluency problems. In a human experiment, we show that a machine-generated sequence outperforms both a random sequence and a sequence generated by a human domain expert. Interestingly, the machine-generated sequence resulted in significantly lower accuracy during training, but higher posttest accuracy. This suggests that the machine-generated sequence induced desirable difficulties. To our knowledge, our study is the first to show that an educational data mining approach can induce desirable difficulties for perceptual learning.

Keywords

visuals, perceptual fluency, implicit learning, desirable difficulties, machine learning, machine teaching, chemistry, optimal training, sequence effects

1. INTRODUCTION

Visual representations are ubiquitous instructional tools in science, technology, engineering, and math (STEM) domains [2, 23]. For example, chemistry instruction on bonding typically includes the visuals shown in Figure 1. While we typically assume that such visuals help students learn because they make abstract concepts more accessible, they can also impede students' learning if students do not know how the visuals show information [27]. To successfully use visuals to learn new domain knowledge, students need representational competencies: knowledge about how visual representations show information [1]. For example, a chemistry student needs to learn that the dots in the Lewis structure in Figure 1(a) show electrons and that the spheres in the space-filling model in Figure 1(b) show regions where electrons likely reside.

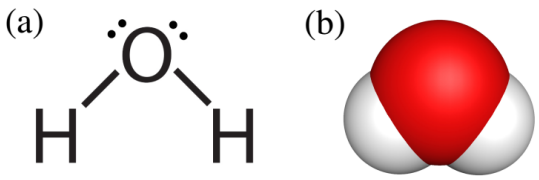

Figure 1: Two commonly used visual representations of water (a: Lewis structure; b: space-filling model).

Most instructional interventions that help students acquire representational competencies focus on *conceptual* representational competencies. These include the ability to map visual features to concepts, support conceptual reasoning with visuals, and choose appropriate visuals to illustrate a given concept [5]. For example, chemists can explain how the number of lines and dots shown in the Lewis structure relate to the colored spheres in the space-filling model by relating these visual features to chemical bonding concepts. Such conceptual representational competencies are acquired via explicit, verbally mediated learning processes that are best supported by prompting students to explain how visuals show concepts [20, 27].

Less research has focused on a second type of representational competency — *perceptual fluency*. It involves the ability to rapidly and effortlessly see meaningful information in visual representations [12, 14]. For example, chemists immediately see that both visuals in Figure 1 show water without having to effortfully think about what the visual shows. They are as fluent at seeing meaning in multiple visuals as bilinguals are fluent in hearing meaning in multiple languages. Perceptual fluency frees up cognitive resources for higher-order complex reasoning, thereby allowing students to use visuals to learn new domain knowledge [16, 27].

Students acquire perceptual fluency via implicit inductive processes [12, 14]. These processes are nonverbal because

verbal reasoning is not necessary [19] and may even interfere with the acquisition of perceptual fluency [20]. Consequently, instructional problems that enhance perceptual fluency engage students in simple problems to quickly judge what a visual shows [19]. For example, one type of perceptual-fluency problem may ask students to quickly and intuitively judge whether two visuals like the ones in Figure 1 show the same molecule. They ask students to rely on implicit intuitions when responding to a series of perceptual-fluency problems. Students typically receive numerous perceptual-fluency problems in a row. The problem sequence is typically chosen so that (1) students are exposed to a variety of visuals and (2) consecutive visuals vary incidental features while drawing students' attention to conceptually relevant features [19, 27].

However, these general principles are underspecified in the sense that they leave room for many possible problem sequences. To date, we lack a principled approach capable of identifying sequences of visual representations that yield optimal learning outcomes for perceptual-fluency problems. To address this issue, we developed a novel educational data mining approach. Using data from human students who learned with perceptual-fluency problems, we trained a machine learning algorithm to mimic human perceptual learning. Then, we used an algorithm to search over possible sequences of visual representations to identify the sequence that was most effective for a machine learning algorithm. In a human experiment, we then tested whether (1) the machine-selected sequence of visual representations yielded higher learning outcomes compared to (2) a random sequence and (3) a sequence generated by a human expert based on perceptual learning principles.

In the following, we first review relevant literature on learning with visual representations, perceptual fluency, and our machine learning paradigm. Then, we describe the methods we used to identify the machine-selected sequence and the methods for the human experiment. We also discuss how our results may guide educational interventions for representational competencies and educational data mining more broadly.

2. PRIOR RESEARCH

2.1 Learning with Visual Representations

Theories of learning with visual representations define visual representations as a specific type of external representation. External representations are objects that stand for something other than themselves — a referent [25]. When we see an image of a pizza, for example, the referent could be a slice of pizza (a concrete object). Alternatively, when used in the context of math instruction, the referent could be a fraction of a whole pizza (an abstract concept). Representations used in instructional materials are defined as external representations because they are external to the viewer. By contrast, internal representations are mental objects that students can imagine and mentally manipulate. Internal representations are the building blocks of mental models; these models constitute students' content knowledge of a particular topic or domain. External representations can be symbolic or visual. For instance, text or equations are symbolic external representations that consist of symbols that have arbitrary (or convention-based) mappings to the referent [32]. By contrast, *visual representations* have similarity-based mappings to the referent [32].

Several theories describe how students learn from visual representations. Mayer's [22] Cognitive Theory of Multimedia Learning (CTML) and Schnotz's [32] Integrated Model of Text and Picture Comprehension (ITPC) draw on information processing theory [4] to describe learning from external representations as the integration of new information into a mental model of the domain knowledge. Here, we focus on learning processes relevant to visual representations.

First, students select relevant *sensory information* from the visual representations for further processing in working memory. To this end, students use perceptual processes that capture visuo-spatial patterns of the representation in working memory [32]. To willfully direct their attention to relevant visual features, students draw on conceptual competencies that enable top-down thematic selection of visual features [15, 17].

Second, students *organize* this information into an internal representation that describes or depicts the information presented in the external representation. Because visual representations have similarity-based analog mappings to referents, their structure can be directly mapped to the analog internal representations [10, 32]. In forming the internal representation, students engage perceptual processes that draw on pattern recognition of objects based on visual cues. They engage conceptual processes to map the visual cues to conceptual representational competencies that allow the retrieval of concepts associated with these objects. The resulting internal representation is a perceptual analog of the visual representation. It is depictive in that its organization directly corresponds to the visuo-spatial organization of the external visual representation [32].

Third, students integrate the information contained in the internal representations into a *mental model* of the domain knowledge (e.g., schemas, category knowledge). To this end, students integrate the analog internal representation into a mental model by mapping the analog features to information in long-term memory. This third step is what constitutes learning: students learn by integrating internal representations into a coherent mental model of the domain knowledge [22, 32, 37].

In sum, students' learning from visual representations hinges on their ability to form accurate internal representations of the representations' referents and on their ability to integrate internal representations into a coherent mental model of the domain knowledge. This process involves both conceptual and perceptual competencies [27]. Although it is well established that conceptual and perceptual competencies are interrelated [16, 17], it makes sense to distinguish them because they are acquired via qualitatively different learning processes [16, 19, 20]. As mentioned earlier, conceptual representational competencies are acquired via verbally mediated, explicit processes [20, 27]. By contrast, perceptual fluency is acquired via implicit, mostly nonverbal processes. Whereas most prior research on instructional interventions for representational competencies has focused on conceptual processes, we focus on perceptual processes.

2.2 Perceptual Fluency

Research on perceptual fluency is based on findings that experts can automatically see meaningful connections among representations, that it takes them little cognitive effort to translate among representations, and that they can quickly and effortlessly integrate information distributed across representations [12]. For example, experts can see "at a glance" that the Lewis structure in Figure 1(a) shows the same molecule as the space-filling model in Figure 1(b). Such perceptual fluency frees cognitive resources for explanation-based reasoning [14, 31] and is considered an important goal in STEM education.

According to the CTML and the ITCP, perceptual fluency involves efficient formation of accurate internal representations of visual representations [22, 32]. Perceptual fluency also involves the ability to combine information from different visual representations without any perceived mental effort and to quickly translate among them [7] [19]. According to the CTML and ITCP, this allows students to map analog internal representations of multiple visual representations to one another [22, 32].

Cognitive science literature [12, 15, 20] suggests that students acquire perceptual fluency via perceptual-induction processes. These processes are inductive because students can infer how visual features map to concepts through experience with many examples [12,15,19]. Students gain *efficiency* in seeing meaning in visuals via perceptual chunking. Rather than mapping specific analog features to concepts, students learn to treat each analog visual as one perceptual chunk that relates to multiple concepts. Perceptual-induction processes are thought to be nonverbal because they do not require explicit reasoning [20]. They are implicit because they occur unintentionally and sometimes unconsciously [33].

Interventions that target perceptual fluency are relatively novel. Kellman and colleagues [19] developed interventions that engage students in perceptual-induction processes by exposing them to many short problems where they have to rapidly translate between representations. For example, students might receive numerous problems that ask them to judge whether two visuals like the ones shown in Figure 1 show the same molecule. These interventions have enhanced students' learning in domains like chemistry [30, 36].

Perceptual learning is strongly affected by problem sequences [27]. To design appropriate problem sequences, consecutive problems expose students to systematic variation (often in the form of contrasting cases) so that irrelevant features vary but relevant features appear across several problems [19]. However, a vital issue remains when designing problem sequences for perceptual-fluency problems: Visual representations differ on a large number of visual features. Consequently, countless potential problem sequences exist that systematically vary these visual features. How do we know which sequence is most effective? To address this issue, we propose a new educational data mining approach that draws on Zhu's machine-teaching paradigm [38, 39]

2.3 Machine Teaching Paradigm

Simply put, machine teaching is the inverse problem of machine learning. Machine learning refers to computer algorithms that select an optimal model for a given set of data. In other words, it determines which model fits the data best. Machine teaching, on the other hand, finds the optimal (smallest) set of data for training such that a given algorithm selects a target model. Although the machine teaching paradigm has been applied to cognitive psychology and education [24], it has not yet been used in educational data mining research.

Machine teaching requires a cognitive model i.e.,a learning algorithm that mimics how human students learn a mapping between visual representations like the ones shown in Figure 1). Given the cognitive model, machine teaching seeks a sequence of learning problems (optimal training sequence $\mathcal{O}$) such that when given $\mathcal{O}$, the learning algorithm learns the mapping. Here, $\mathcal{O}$ need not be independent and identically distributed (i.i.d.). Machine teaching can be viewed as a communication problem between a teacher and a student: The goal is to communicate the mapping using the shortest message. The channel only allows messages in the form of a training sequence and the student decodes the message with the learning algorithm. In perceptual learning, students learn a mapping between visual features of two types of visual representations, allowing them to fluently translate among the visual representations.

To evaluate whether a training sequence is effective, we test the cognitive model's performance at mapping visual representations using a different set of perceptual-fluency problems than used during training. Typically, a sequence of training problems (aka training instances in machine learning) is drawn from a distribution of perceptual-fluency problems used for training (P_t). The set of test problems comes from a separate distribution of perceptual-fluency problems (P_e). The goal is to minimize the test error rate on P_e. The goal of machine teaching then becomes:

$$\mathcal{O} = \underset{S \in \mathcal{C}_t}{\operatorname{argmin}} P_{(x,y)\sim P_e} \left(\mathcal{A}(S)(x) \neq y\right) \tag{1}$$

Here, $\mathcal{C}_t$ is the set of all possible training sequences and $\mathcal{A}(S)$ is the learned hypothesis after training on the sequence S. Note that, $\mathcal{O}$ is not necessarily an i.i.d. sequence drawn from P_t. One practical approach to approximately solve the optimization problem is shown in Algorithm 1. To properly construct the optimal training sequence in this given setting, we must understand:

1. the nature of the to-be-learned domain knowledge
2. the learning algorithm the cognitive model is using

In this paper, the to-be-learned domain knowledge is well-known. It is the mappings between visual representations that students have to learn. Further, we used data from human students learning from perceptual-fluency problems to generate a cognitive model that mimics how humans learn mappings between visual representations. Our goal is to investigate whether, when the mappings and the cognitive

model are well understood, machine teaching can identify a training set that is more effective than (a) a problem sequence based on perceptual learning principles and (b) a random sequence.

Algorithm 1 Machine Teaching

```
Input: Learner 𝒜, Test Distribution P_e
𝒪 ← Starting sequence
ε_best ← error(train(𝒜, 𝒪), P_e)
while TRUE do
  𝒩 ← neighbors(𝒪), ε_old ← ε_best
  for S ∈ 𝒩 do
    ε ← error(train(𝒜, S), P_e)
    if ε < ε_best then
      ε_best ← ε, 𝒪 ← S
    end if
  end for
  if ε_best = ε_old then
    return 𝒪
  end if
end while
```

3. COGNITIVE MODEL

We now describe how we constructed the cognitive model that was used to construct the training sequence. To this end, we first describe the perceptual-fluency problems, then describe how we formally represented these problems, which learning algorithm the cognitive model used, and finally how we used the cognitive model to identify the optimal training sequence.

3.1 Perceptual-Fluency Problems

Perceptual-fluency problems are single-step problems that ask students to make simple perceptual judgments. In our case, students were asked to judge whether two visual representations showed the same molecule, as shown in Figure 2. Students were given two images. One image was of a molecule represented by a Lewis structure and the other image was a molecule represented by a space-filling model. They were asked to judge whether those two images show the same molecule or not.

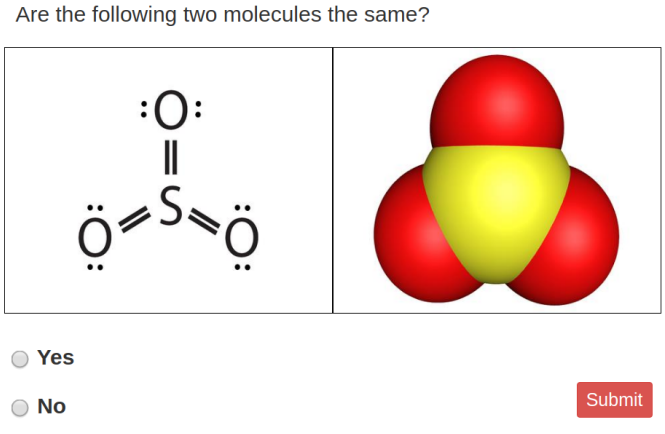

Figure 2: In this sample perceptual-fluency problem, students judged whether or not the Lewis structure and the space-filling model showed the same molecule. The answer is yes.

3.2 Visual Representation of Molecules

In our experiment, we used visual representations of chemical molecules common in undergraduate instruction. To identify these molecules, we reviewed textbooks and web-based instructional materials. We counted the frequency of different molecules using their chemical names (e.g., H_2O) and common names (e.g., water), and chose the 142 most common molecules. In order to formally describe the visual representations, we quantified visual features for each of the molecules. To this end, we first hand-coded the visual features that were present in the visual representations. For Lewis structures, these hand-coded features included counts of individual letters as well as information about different bonds present in each molecule, among others. For space-filling models, hand-coded features included counts of colored spheres, bonds, and other features. Further, we included several surface features that we expect human students attend to based on findings that humans tend to focus on broader surface features that are easily perceivable. Then we used the method found in [29] to determine which subset of features (each for Lewis structure and space-filling model) humans attend to most. Building on these results, we created feature vectors for each of the molecules (Figure 3). These feature vectors of Lewis structures and space-filling models contained 27 and 24 features, respectively. These feature vectors were then used to train and test the learning algorithm.

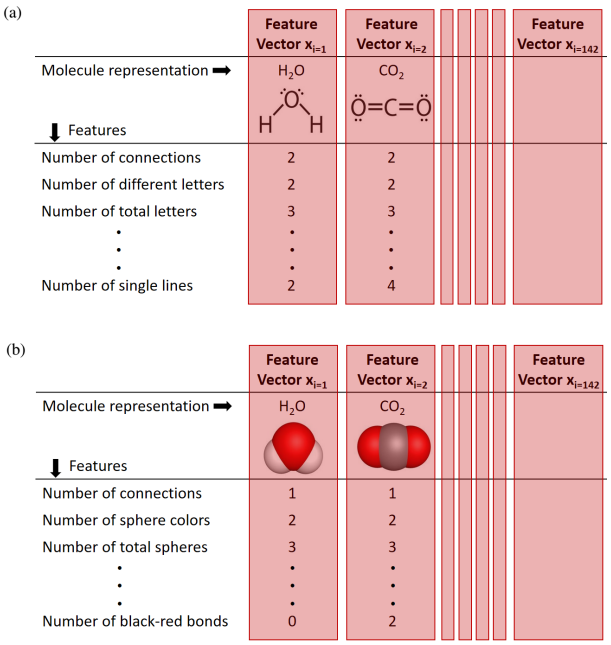

Figure 3: Example features for H_2O and CO_2 molecule representations with feature vectors in red (a: Lewis structure; b: space-filling model).

3.3 Learning Algorithm

We used a feed-forward artificial neural network (ANN) [8] as our learning algorithm. ANN is inspired by the biological neural network. A biological neuron produces an output when collective effect of its inputs reaches a certain threshold. It is still not clear exactly how the human brain learns but one assumption is that it is associated with the interconnection between the neurons. ANNs try to model this

low level functionality of the brain. We chose ANN to be our learning algorithm due to this similarity. Our ANN took two feature vectors (x_1 and x_2) as input. Each feature vector corresponded to one of the two molecules shown. Given this input, the ANN produced a probability that the two molecules were the same. Then, given the correct answer $y \in \{0, 1\}$ (here 1 means the two molecules are the same), the ANN updated its weights using the backpropagation algorithm. The backpropagation algorithm uses gradients to converge to an optima. Algorithm 2 shows the training procedure of the neural network. It shows that the update procedure also used a history window and multiple backpropagation passes, an atypical approach for an ANN. We took two measures to address the issue that regular ANN algorithms do not learn from memory like humans do. First, we assumed that humans remember a fixed number of past consecutive problems. Second, we assumed that after receiving feedback on the latest problem, humans update their internal model by reviewing memorized problems (along with the latest problem) several times. To emulate this behavior, we introduced the history window and multiple backpropagation passes. This procedure was followed for all problems in a given training sequence.

Algorithm 2 train: training method for the NN learner

```
Input: Training sequence S , Learning rate η, History window size w, Number of backpropagations b
H ← [] //initialize an empty history window
for i = 1 → |S| do
    append(H, S[i]) //update history window
    // train on the history window
    w' ← |H|
    for k = 1 → b do
        for j = 1 → w' do
            (x, y) ← H[j]
            backprop(x, y, η)
        end for
    end for
    //check history window size
    if w' > w then
        H.remove(0) //remove the oldest instance in history
    end if
end for
```

A further, structural difference between our learning algorithm from a general artificial neural network is that our learning algorithm had two separate weight columns (one for each representation of the input molecules). The model architecture of the ANN is shown in Figure 4. Here, the weights and outputs from one of the columns did not interact with those of the other column until the output layer. The network mapped the two inputs (feature vectors x_1 and x_2) to a space wherein the same molecule shown by different representations are close to each other while different molecules are distant. These mapping functions are called embedding functions (one for each representation) and the space is called a common embedding space. Once the mapping was complete, a judgment was possible regarding the similarity of the input molecules. This judgment was based on the distance in the common embedding space and made in the output layer of the ANN. Embeddings were generated in the layer before the output layer—the embedding layer.

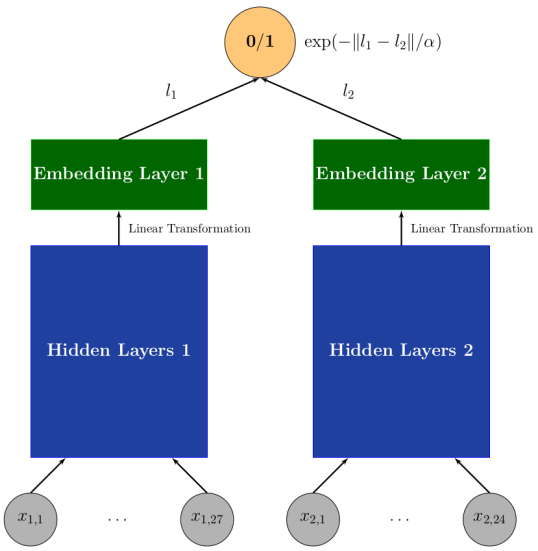

Figure 4: Structure of the Artificial Neural Network learning algorithm

Neurons in an ANN use a non-linear function called activation function to introduce non-linearity. For all hidden layers before the embedding layer, we used the leaky rectifier [21] activation function (the neuron employing leaky rectifier is called a leaky rectified linear unit or leaky ReLU). A standard rectified linear unit (ReLU) allows only positive inputs to move onwards (outputs 0 otherwise). A leaky ReLU, on the other hand, outputs a small scaled input when the input is negative. Both ReLU and leaky ReLU have strong biological motivations. According to cognitive neuroscience studies of human brains, neurons encode information in a sparse and distributed fashion [3]. Using ReLU, ANNs can also encode information sparsely. Besides this biological plausibility, sparsity also confers mathematical benefits like information disentangling and linear separability. Rectified linear units also enable better training of ANNs [13]. The embedding layers, by contrast, do not use activation functions. Hence, the output of embedding layers are a linear transformation of its inputs. Given the inputs (x_1, x_2), let the ANN-generated embeddings be l_1 and l_2, respectively. Then, we computed the probability of the two representations showing the same molecule in the output layer using the following equation:

$$\exp\left(-\frac{\|l_1 - l_2\|}{\alpha}\right) \tag{2}$$

Here, α is a trainable parameter that the ANN learns along with the weights. We thresholded this value at 0.5 to generate the ANN prediction $\hat{y} \in \{0, 1\}$.

3.4 Pilot Study - Train the Learning Algorithm

Our first step was to train the learning algorithm to mimic human perceptual learning. To this end, we conducted a pilot experiment to find a good set of hyperparameters for the ANN learning algorithm. Hyperparameters of an ANN are variables that are set before optimizing the weights (e.g.,

number of hidden layers, number of neurons in each layer, learning rate etc.). Our goal was to identify hyperparameters that make predictions matching human behavior on the posttest. Hence, we matched the algorithm's predictions to summary statistics of human performance on the posttest.

Our pilot experiment included 47 undergraduate chemistry students. They were randomly assigned to one of two conditions that used a random training sequence: supervised training ($n = 35$) or unsupervised training ($n = 12$). Participants in the supervised training condition received feedback after each training problem, whereas participants in the unsupervised condition did not receive feedback. We included the unsupervised training condition to generate an evaluation set (used to determine the success of pretraining). This evaluation set was used to pretrain the ANN learning algorithm.

Let there be n supervised human participants. Each participant received a random pretest set, a random training sequence, and a random posttest set. We trained the ANN learning algorithm n times independently (once for each participant). While training for the i-th time we used the training sequence viewed by the i-th supervised human participant. The same posttest set viewed by this participant was also used to evaluate the performance of the ANN learning algorithm after training. Let the error on this posttest set for the i-th human participant and trained ANN learning algorithm be pp_i and pn_i respectively. Then, Equation 3 is a measure used to determine whether or not an ANN learning algorithm's performance is comparable to the average human. Note that lower *error rates* are desirable.

$$error\ rates = \left| \frac{1}{n} \left(\sum_{i=1}^{n} pp_i - \sum_{i=1}^{n} pn_i \right) \right| \tag{3}$$

Table 1 reports the accuracies of participants in the pilot experiment.

Table 1: Accuracy in Pilot Experiment by Training Condition. Average pretest, training and posttest accuracy with SEM in parentheses.

Condition	Pretest	Training	Posttest
Supervised	79.9 (1.8)	75.7 (1.2)	89.4 (1.4)
Unsupervised	77.9 (3.4)	78.5 (2.8)	77.1 (3.3)

We note that humans usually have some degree of prior knowledge about chemistry. By contrast, the weights of an ANN are generally initialized at random. We address this issue by modeling the effect of prior knowledge, specifically we introduced a pretraining phase for the ANN learning algorithm. To this end, we drew a large sample of instances (10000) from the combined test and training distribution ($\frac{1}{2}P_e + \frac{1}{2}P_t$) to form a pretraining set. Further, we combined the pretest problem across both the supervised and unsupervised conditions, along with the training problems in the unsupervised condition to form the pretraining evaluation set. Because we did not provide feedback for these problems, we assumed that the participants did not learn anything new while going through them. Formally, let participants' error on the pretraining evaluation set be called human pretraining error. We then trained the ANN learning algorithm on the pretraining set. Note that an ANN can train over the over the same set over multiple iterations (formally known as epochs). We trained the ANN learning algorithm until its error on the pretraining evaluation set was smaller than human pretraining error. This concluded the pretraining phase.

We used standard coordinate descent with random restart to find a good hyperparameter set. Coordinate descent successively minimizes the *error rates* along the coordinate directions (e.g., embedding size, learning rate). At each iteration, the algorithm chooses one particular coordinate direction while fixing the other values. Then, it minimizes in the chosen coordinate direction. Table 2 shows the values of the hyperparameters over which we decided to explore along with the best value found. These hyperparameters were used to identify the optimal training sequence.

3.5 Finding an Optimal Training Sequence

We used the ANN learning algorithm to generate an optimal training sequence for the perceptual-fluency problems. In Equation 1, we defined the optimization problem to solve. We solved this problem by searching over the space of all possible training sequences. Without limiting the size of the training sequence, the search space becomes infinite and infeasible. To mitigate this issue, we set the size of the candidate training sequences to 60. This aligns with prior research on perceptual learning [28]:

$$\mathcal{O} = \underset{S \in \mathcal{C}_t, |S|=60}{\operatorname{argmin}} P_{(x,y)\sim P_e} \left(\mathcal{A}(S)(x) \neq y \right) \tag{4}$$

We used a modified hill climbing algorithm to find such an optimal training sequence. Hill climb search takes a greedy approach. Procedurally, we started with one particular training sequence. Then, we evaluated neighbors of that particular training sequence to determine whether a better one existed. If so, we moved to that one. This process stopped when no such neighbors were found. This search algorithm is defined with its states and neighborhood definition:

- **States:** Any training sequence $S \in \mathcal{C}_t$ of size 60
- **Initial State:** A training sequence selected by a domain expert.
- **Neighborhood of S:** Any training sequence that differs with S by one problem is a neighbor. For computational efficiency, we restricted ourselves to only inspecting 500 neighbors for a given training sequence. We do so by first selecting a problem S uniformly at random. Then we replace the selected problem with 500 randomly selected problems with the same answer (i.e., same y value). This made our search algorithm stochastic.

4. HUMAN EXPERIMENT

Our main goal was to evaluate whether the optimal training sequence yields higher learning outcomes. To this end, we conducted a randomized, controlled experiment with humans. Here, we discuss our experimental setup and associated results.

Table 2: Hyper-parameters for the ANN learning algorithm

Parameter name	Values explored	Best value
Embedding size	$1, 2, 4, 8, 16$	2
Learning rate	$0.00001, 0.00005, 0.0001, 0.0005, 0.001, 0.005, 0.01, 0.05, 0.1$	0.0001
History window size	$0, 1, 2, 4, 8, 16, 32, 60$	2
Backprop count	$1, 2, 4, 8, 16$	2
Number of hidden layers before embedding layer	$0, 1, 2, 3, 4$	0
Number of hidden units in each column	$10, 20, 40, 80, 160$	N/A

4.1 Participants

We recruited 368 participants using Amazon's Mechanical Turk (MTurk) [6]. Among them, 216 were male and 131 were female. The rest did not disclose their gender. Most of the participants were below the age of 45 (86%) and the greatest number (192) fell in the age group $24-35$. Among the 95.4% who disclosed their knowledge about chemistry, 45.7% had taken an undergraduate-level chemistry class.

4.2 Test Set

Because our goal was to assess transfer of learning from the training sequence to a novel test set, we chose training and test problems from separate distributions. Hence, we randomly divided the 142 molecules that we selected for this experiment into two sets of 71 (training molecules, $\mathcal{X}_t$ and test molecules $\mathcal{X}_e$). One of the sets was used to create the test distribution, whereas the other one was used to create the training distribution. We now describe in more detail how we created the test distribution P_e because our goal was to reduce humans' error rates on the test set. We used the following procedure.

- $x_1 \sim p_1$, where p_1 is a marginal distribution on $\mathcal{X}_e$. p_1 is "importance of molecule x_1 to chemistry education" and was constructed by manually searching a corpus of chemistry education articles for molecule text frequency.
- With probability $1/2$, set $x_2 = x_1$ so that the true answer $y = 1$.
- Otherwise, draw $x_2 \sim p_2(\cdot \mid x_1)$. The conditional distribution p_2 is based on domain experts' opinion that favors confusable x_1, x_2 pairs in an education setting. Also note that, $p_2(x_1|x_1) = 0, \forall x_1$. Taken together,

$$P_e(x_1, x_2) = \frac{1}{2} p_1(x_1) \mathbb{I}_{\{x_1 = x_2\}} + \frac{1}{2} p_1(x_1) p_2(x_2 \mid x_1).$$

Both the pretest and posttest judgment problems were sampled from this distribution across all conditions.

4.3 Experimental Design

We compared three training conditions:

1. In the *machine training sequence* condition, we used the optimal training sequence $\mathcal{O}$ found by the modified hill climb search algorithm. For all $(x_1, x_2) \in \mathcal{O}$ (here $x_1 \in \mathcal{X}_t, x_2 \in \mathcal{X}_t$), the corresponding true answer y was the indicator variable on whether x_1 and x_2 were the same molecule: $y = \mathbb{I}_{\{x_1 = x_2\}}$. We presented x_1 and x_2 in Lewis and space-filling representations to the human participants, respectively. Participants gave their binary judgment $\hat{y} \in \{0, 1\}$. We then provided the true answer y as feedback to the participant.
2. In the *human training sequence* condition, the training sequence was constructed by a domain expert using perceptual learning principles (using molecules only from $\mathcal{X}_t$). Specifically, an expert on perceptual learning constructed the sequence based on the contrasting cases principle [19, 30], so that consecutive examples emphasized conceptually meaningful visual features, such as the color of spheres that show atom identity or the number of dots that show electrons. The rest of this condition was the same as the machine training sequence condition. This training sequence is identical to the initial state of the modified hill climb search algorithm that we used to generate the machine training sequence.
3. In the *random training sequence* condition, each training problem (x_1, x_2) was selected from the training distribution P_t with $y = \mathbb{I}_{\{x_1 = x_2\}}$. The training distribution P_t for this condition was induced in the same manner as the test distribution P_e but on the set of training molecules $\mathcal{X}_t$. The rest of the condition was the same as the previous ones.

4.4 Procedure

We hosted the experiment on the Qualtrics survey platform [26] using NEXT [18]. Participants first received a brief description of the study and then completed a sequence of 126 judgment problems (yes or no). The problems were divided into three phases as follows. First, participants received a pretest that included 20 test problems without feedback. Second, participant received the training, which included 60 training problems sequenced in correspondence to their experimental condition. During this phase, correctness feedback was provided for submitted answers. We assumed that participants learned during this phase because they received feedback. Third, participants received a posttest that included 40 test problems without feedback. In addition, one *guard problem* was inserted after every 19 problems throughout all three phases. A guard question either showed two identical molecules depicted by the same representation or two highly dissimilar molecules depicted by Lewis structures. We used these guard questions to filter out participates who clicked through the problems haphazardly. In our main analyses, we disregarded the guard problems. So that no visual representation was privileged, we randomized their positions (left vs. right).

4.5 Results

Of the 368 participants, we excluded 43 participants who failed any of the guard questions. The final sample size was $N = 325$. The final number of participants in the conditions random, human, and machine training sequence were 108, 117 and 100 respectively. Table 3 reports accuracy on the pretest, training set, and posttest. See Figure 5 for a graphical depiction of the same data.

Table 3: Accuracy by Training Condition. Average pretest, training and posttest accuracy with SEM in parentheses.

Condition	Pretest	Training	Posttest
Machine	69.5 (1.1)	63.9 (1.1)	74.7 (1.1)
Human	71.3 (1.3)	72.4 (1.0)	71.7 (1.0)
Random	69.4 (1.1)	70.3 (1.1)	71.1 (1.1)

4.5.1 *Effects of condition on training accuracy*

First, we tested whether training condition affected participants' accuracy during training. To this end, we used an ANCOVA (Analysis of COVAriance) with condition as the independent factor and training accuracy as the dependent variable. Because pretest accuracy was a significant predictor of training accuracy, we included pretest accuracy as the covariate. Results showed a significant main effect of condition on training accuracy, $F(2, 321) = 18.8, p < .001, \eta^2 = .082$. Tukey post-hoc comparisons revealed that (a) the machine training sequence condition had significantly lower training accuracy than the human training sequence condition ($p < .001, d = -0.32$), (b) the machine training sequence condition had significantly lower training accuracy than the random training sequence condition ($p < .001, d = -0.26$), and (c) no significant differences existed between the human and random training sequence conditions ($p = .592, d = 0.05$). In other words, during the training phase, the human and random training sequences were equally effective in terms of accuracy, but the machine training sequence was less effective.

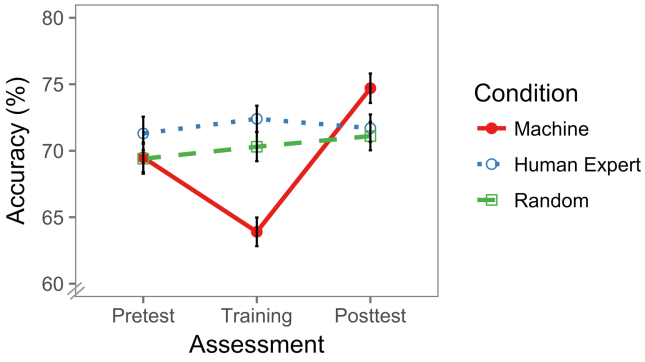

Figure 5: Learning progress between conditions revealed an initial disadvantage, but ultimate advantage for the machine-generated sequence.

4.5.2 *Effects of condition on posttest accuracy*

Next, we tested whether training condition affected participants' posttest accuracy. To this end, we conducted an ANCOVA with condition as the independent factor and posttest accuracy as the dependent variable. Because pretest accuracy was a significant predictor of posttest accuracy, we included pretest accuracy as a covariate. Results showed a significant main effect of condition on posttest accuracy, $F(2, 321) = 5.02, p < .01, \eta^2 = .023$. Tukey post-hoc comparisons revealed that (a) the machine training sequence condition had significantly higher posttest accuracy than the human training sequence condition ($p < .05, d = 0.16$), (b) the machine training sequence condition had significantly higher posttest accuracy than the random sequence condition ($p < .05, d = 0.14$), and (c) no significant differences existed between the human and random training sequence conditions ($p = .960, d = -0.02$). In other words, the human and random training sequences were equally effective and the machine training sequence was most effective.

5. DISCUSSION

Our goal was to investigate whether a novel educational data mining approach can help identify a training sequence of visual representations that enhances students' learning from perceptual-fluency problems. To this end, we applied the machine teaching paradigm. It involved gathering data from human students learning from perceptual-fluency problems. Next, we generated a cognitive model that mimics human perceptual learning. We then used the cognitive model to reverse-engineer an optimal training sequence for a machine-learning algorithm. Finally, we conducted an experiment that compared the machine training sequence to a random sequence and to a principled sequence generated by a human expert on perceptual learning. Results showed that the machine training sequence resulted in lower performance during training, but greater performance on a posttest.

These findings make several important contributions to the perceptual learning literature. First, our results can inform the instructional design of perceptual-learning problems. Even though prior research yields principles for effective sequences of visual representations, numerous potential sequences can satisfy these principles. Our results show that this new educational data mining approach can help address this problem. Given a learning algorithm that constitutes a cognitive model of students learning a task, instructors can identify a sequence of problems that likely yields higher learning outcomes.

Second, our results expand theory on perceptual learning. The fact that the machine learning sequence yielded lower performance during training but greater posttest scores suggests that this sequence induced desirable difficulties during learning [19, 34, 40]. The concept of desirable difficulties describes the common finding that instructional techniques yield lower performance during training, but higher long-term learning outcomes. To explain this phenomenon, Soderstrom and Bjork [34] proposed that more difficult learning interventions induce more active processing during training. This lowers immediate performance due to the increased difficulty, but results in more durable memories and greater long-term learning. Our findings suggest that the machine teaching approach was successful because it identified a training sequence that induced desirable difficulties. To the best of our knowledge, our study is the first to show that an educational data mining approach can be used to induce desirable difficulties for perceptual learning.

Our findings also contribute to the educational data mining literature. We provide the first empirical evidence that

an ANN learning algorithm constitutes an adequate cognitive model of learning with visual representations. As far as we know, the machine teaching paradigm has thus far only been applied to learning with artificial visual stimuli that vary on only one or two dimensions (e.g. Gabor patches [11]). Thus, our study provides the first demonstration that machine learning along with machine teaching is a viable approach to modeling and improving learning with realistic, high-dimensional visual representations like Lewis structures and space-filling models of chemical molecules. Many other domains like biology, engineering, math also use high-dimensional visual representations. Therefore, we believe this approach is valuable for educational data mining research.

6. LIMITATIONS AND FUTURE DIRECTIONS

Our findings should be interpreted against the background of the following limitations. First, the population of MTurk workers may limit generalization to the target population of undergraduate chemistry students. MTurk workers have highly variable prior knowledge about chemistry. As mentioned previously, around 45.7% of the participants had taken an undergraduate level chemistry class. This suggests that their prior knowledge may have been lower and more diverse than that of a typical undergraduate chemistry student. Hence, we plan to test whether the machine training sequence leads to better learning for undergraduate chemistry students.

Second, the search algorithm we used to find the machine training sequence did not test all possible training sequences of size 60. As mentioned previously, we only inspected 500 neighbors (out of a potential $5040 = 71 \times 71 - 1$) for any given training sequence. Moreover, we stopped the search algorithm after a predetermined amount of time. We chose this inexhaustive approach because exhaustively finding a solution is not computationally feasible. Thus, we settled for a suboptimal training sequence that still yielded a small risk on the test distribution. Consequently, it is possible to find a better training sequence than the one we used in our experiments.

Third, while determining the hyperparameters of the ANN learning algorithm such that it mimics human perceptual learning, we only searched over a subset of all possible hyperparameters. As a result, it is possible that a better set of hyperparameters exists. Our study was also limited in that we did not account for individual prior knowledge. Hence, future research needs to investigate how to expand the approach presented in this paper to modeling individual prior knowledge (e.g., for adaptive teaching or personal training).

A fourth limitation of the present experiments is that our study was constrained in the use of chemistry representations as stimuli. While we used realistic representations that are more high-dimensional than prior perceptual learning studies [9, 11, 35] and that are more representative of commonly used visual representations in a variety of STEM domains, the complexity of the representations we considered does not reflect all realistic stimuli. Still we see no reason why this approach could not be applied to other representations in other domains. Sparser and richer visuals exist and it is possible that machine teaching may yield greater benefits for sparser visuals. We will investigate this hypothesis in future studies.

7. CONCLUSION

This paper advanced a novel educational data mining approach to identify optimal sequences of visual representations for perceptual-fluency problems. Students' difficulties in learning with visual representations is partly due to a lack of perceptual fluency. This increases the cognitive demands of learning with visuals. Perceptual-fluency problems are a relatively novel type of instructional intervention that can aid learning from visuals by freeing up cognitive resources for higher-order complex reasoning. Thus far, we have lacked a principled approach capable of identifying effective sequences of visual representations. Our educational data mining approach relied solely on students' responses to perceptual-fluency problems to select a sequence of visuals that is effective for a machine learning algorithm mimicking human perceptual learning. Our results showed that this approach is more effective than conventional perceptual fluency instruction. Further, the effectiveness of our approach lies in its ability to induce desirable difficulties. Given the pervasiveness of visual representations in STEM domains, we anticipate that our findings will be broadly useful for students' learning with visual representations. We also plan to investigate how the machine generated sequence induced desirable difficulties in the humans.

Acknowledgement

This is supported in part by NSF grant IIS 1623605. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the NSF. We also thank Michael Mozer and Brett Roads (University of Colorado, Boulder) for their support and comments regarding the cognitive model.

8. REFERENCES

[1] Ainsworth, S. DeFT: A conceptual framework for considering learning with multiple representations. *Learning and instruction 16*, 3 (2006), 183–198.

[2] Ainsworth, S. The educational value of multiple-representations when learning complex scientific concepts. *Visualization: Theory and practice in science education* (2008), 191–208.

[3] Attwell, D., and Laughlin, S. B. An energy budget for signaling in the grey matter of the brain. *Journal of Cerebral Blood Flow & Metabolism 21*, 10 (2001), 1133–1145.

[4] Baddeley, A. Working memory. *Science 255*, 5044 (1992), 556–559.

[5] Bodemer, D., Ploetzner, R., Feuerlein, I., and Spada, H. The active integration of information during learning with dynamic and interactive visualisations. *Learning and Instruction 14*, 3 (2004), 325–341.

[6] Buhrmester, M., Kwang, T., and Gosling, S. D. Amazon's mechanical turk: A new source of inexpensive, yet high-quality, data? *Perspectives on psychological science 6*, 1 (2011), 3–5.

[7] Chase, W. G., and Simon, H. A. Perception in chess. *Cognitive psychology 4*, 1 (1973), 55–81.

[8] Demuth, H. B., Beale, M. H., De Jess, O., and Hagan, M. T. *Neural network design*. Martin Hagan, 2014.

[9] Eilam, B. *Teaching, learning, and visual literacy: The dual role of visual representation*. Cambridge University Press, 2012.

[10] Gentner, D., and Markman, A. B. Structure mapping in analogy and similarity. *American psychologist 52*, 1 (1997), 45.

[11] Gibson, B. R., Rogers, T. T., Kalish, C., and Zhu, X. What causes category-shifting in human semi-supervised learning? In *CogSci* (2015).

[12] Gibson, E. J. Perceptual learning in development: Some basic concepts. *Ecological Psychology 12*, 4 (2000), 295–302.

[13] Glorot, X., Bordes, A., and Bengio, Y. Deep sparse rectifier neural networks. In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics* (2011), pp. 315–323.

[14] Goldstone, R. L., and Barsalou, L. W. Reuniting perception and conception. *Cognition 65*, 2 (1998), 231–262.

[15] Goldstone, R. L., Medin, D. L., and Schyns, P. G. *Perceptual learning*. Academic Press, 1997.

[16] Goldstone, R. L., Schyns, P. G., and Medin, D. L. Learning to bridge between perception and cognition. *The psychology of learning and motivation 36* (1997), 1–14.

[17] Harel, A. What is special about expertise? visual expertise reveals the interactive nature of real-world object recognition. *Neuropsychologia 83* (2016), 88–99.

[18] Jamieson, K. G., Jain, L., Fernandez, C., Glattard, N. J., and Nowak, R. Next: A system for real-world development, evaluation, and application of active learning. In *Advances in Neural Information Processing Systems* (2015), pp. 2656–2664.

[19] Kellman, P. J., and Massey, C. M. Perceptual learning, cognition, and expertise. *The psychology of learning and motivation 58* (2013), 117–165.

[20] Koedinger, K. R., Corbett, A. T., and Perfetti, C. The knowledge-learning-instruction framework: Bridging the science-practice chasm to enhance robust student learning. *Cognitive science 36*, 5 (2012), 757–798.

[21] Maas, A. L., Hannun, A. Y., and Ng, A. Y. Rectifier nonlinearities improve neural network acoustic models. In *Proc. icml* (2013), vol. 30, p. 3.

[22] Mayer, R. E. *Cognitive theory of multimedia learning*. The cambridge handbook of multimedia learning (2nd ed., pp. 31-48). New York, NY: Cambridge University Press, 2009.

[23] NRC. *Learning to Think Spatially*. Washington, D.C.: National Academies Press, 2006.

[24] Patil, K. R., Zhu, X., Kopeć, Ł., and Love, B. C. Optimal teaching for limited-capacity human learners. In *Advances in neural information processing systems* (2014), pp. 2465–2473.

[25] Peirce, C. S., Hartshorne, C., and Weiss, P. *Collected Papers of Charles Sanders Peirce: (Vol. I-VI)*. MA: Harvard University Press, 1935.

[26] Qualtrics. Qualtrics©2018. `https://it.wisc.edu/services/surveys-qualtrics`, last visited 01-18-2018, 2005.

[27] Rau, M. A. Conditions for the effectiveness of multiple visual representations in enhancing stem learning. *Educational Psychology Review 29*, 4 (2017), 717–761.

[28] Rau, M. A., Aleven, V., and Rummel, N. Successful learning with multiple graphical representations and self-explanation prompts. *Journal of Educational Psychology 107*, 1 (2015), 30.

[29] Rau, M. A., Mason, B., and Nowak, R. D. How to model implicit knowledge? similarity learning methods to assess perceptions of visual representations. In *Educational Data Mining* (2016), pp. 199–206.

[30] Rau, M. A., Michaelis, J. E., and Fay, N. Connection making between multiple graphical representations: A multi-methods approach for domain-specific grounding of an intelligent tutoring system for chemistry. *Computers & Education 82* (2015), 460–485.

[31] Richman, H. B., Gobet, F., Staszewski, J. J., and Simon, H. A. Perceptual and memory processes in the acquisition of expert performance: The epam model. *The road to excellence: The acquisition of expert performance in the arts and sciences, sports, and games* (1996), 167–187.

[32] Schnotz, W. An integrated model of text and picture comprehension. *The Cambridge handbook of multimedia learning (2 ed., pp. 72-103)* (2014).

[33] Shanks, D. R. Implicit learning. *Handbook of cognition* (2005), 202–220.

[34] Soderstrom, N. C., and Bjork, R. A. Learning versus performance: An integrative review. *Perspectives on Psychological Science 10*, 2 (2015), 176–199.

[35] Uttal, D. H., and Doherty, K. O. Comprehending and learning from 'visualizations': A developmental perspective. *Visualization: Theory and practice in science education* (2008), 53–72.

[36] Wise, J. A., Kubose, T., Chang, N., Russell, A., and Kellman, P. J. Perceptual learning modules in mathematics and science instruction. *Teaching and learning in a network world'(IOS Press, 2000)* (2003), 169–176.

[37] Wylie, R., and Chi, M. T. The self-explanation principle in multimedia learning. *R. E. Mayer (Ed.), The Cambridge handbook of multimedia learning* (2014), 413–432.

[38] Zhu, X. Machine teaching: An inverse problem to machine learning and an approach toward optimal education. In *AAAI* (2015), pp. 4083–4087.

[39] Zhu, X., Singla, A., Zilles, S., and Rafferty, A. N. An overview of machine teaching. *arXiv preprint arXiv:1801.05927* (2018).

[40] Ziegler, E., and Stern, E. Delayed benefits of learning elementary algebraic transformations through contrasted comparisons. *Learning and Instruction 33* (2014), 131–146.