An Empirical Study of Non-Expert Curriculum Design for Machine Learners

Bei Peng
Washington State University
beipeng.peng@gmail.com

James MacGlashan
Brown University
jmacglashan@gmail.com

Robert Loftin
North Carolina State University
rtloftin@csc.ncsu.edu

Michael L. Littman
Brown University
mlittman@cs.brown.edu

David L. Roberts
North Carolina State University
robertsd@csc.ncsu.edu

Matthew E. Taylor
Washington State University
taylorm@eecs.wsu.edu

Abstract

Existing machine-learning work has shown that algorithms can benefit from *curriculum learning*, a strategy where the target behavior of the learner is changed over time. However, most existing work focuses on developing automatic methods to iteratively select training examples with increasing difficulty tailored to the current ability of the learner, neglecting how non-expert humans may design curricula. In this work we introduce a curriculum-design problem in the context of reinforcement learning and conduct a user study to explicitly explore how non-expert humans go about assembling curricula. We present results from 80 participants on Amazon Mechanical Turk that show 1) humans can successfully design curricula that gradually introduce more complex concepts to the agent within each curriculum, and even across different curricula, and 2) users choose to add task complexity in different ways and follow salient principles when selecting tasks into the curriculum. This work serves as an important first step towards better integration of non-expert humans into the reinforcement learning process and the development of new machine learning algorithms to accommodate human teaching strategies.

1 Introduction

Humans acquire knowledge efficiently through a highly organized education system, starting from simple concepts, and then gradually generalizing to more complex ones using previously learned information. Similar ideas are exploited in animal training [Skinner, 1958]—animals can learn much better through progressive task shaping. Recent work [Bengio *et al.*, 2009; Kumar *et al.*, 2010; Lee and Grauman, 2011] has shown that machine learning algorithms can benefit from a similar training strategy, called *curriculum learning*. Rather than considering all training examples at once, the training data can be introduced in a meaningful order based on their apparent simplicity to the learner, such that the learner can build up a more complex model step by step. The agent will be able to learn faster on more difficult examples after it has learned on simpler examples. This training strategy was shown to drastically affect learning speed and generalization [Bengio *et al.*, 2009; Kumar *et al.*, 2010].

While most existing work on curriculum learning (in the context of machine learning) focuses on developing an automatic method to iteratively select training examples with increasing difficulty tailored to the current ability of the learner [Kumar *et al.*, 2010; Lee and Grauman, 2011], how *humans* design curricula is one neglected topic. A better understanding of the curriculum design strategies used by humans may lead to the development of new machine learning algorithms that accommodate human teaching strategies. Another motivation for this work is the increasing need for non-expert humans to teach autonomous agents new skills without programming. A number of published works in Interactive Reinforcement Learning [Thomaz and Breazeal, 2006; Knox and Stone, 2009; Griffith *et al.*, 2013] has shown that reinforcement learning (RL) [Sutton and Barto, 1998] agents can successfully speed up learning using human feedback, demonstrating the significant role humans play in teaching an agent to learn a (near-) optimal policy. Taylor first proposed that curricula should be automatically designed in an RL context, and that we should try to leverage human knowledge to design more efficient curricula [2009]. As more robots and virtual agents become deployed, the majority of teachers will be non-experts. This work focuses on understanding non-expert human teachers rather than finding the most efficient way to solve our RL problem—future work will investigate how to adapt machine learning algorithms to better take advantage of this type of non-expert curricula. We believe this work is the first to explicitly study how non-expert humans approach designing curricula for RL domains.

We are interested in studying whether humans can identify the concepts an agent needs to learn in the curriculum to complete a given target task. Given that humans can arbitrarily select a sequence of tasks with different level of complexities, we hypothesize that humans gradually introduce more complex concepts to the agent within each curriculum. It is interesting to explore how humans increase task complexity and general principles regarding efficient curricula by analyzing the humans' design processes. If we can discover salient patterns within the curricula, we may be able to automate the *active selection* of suitable tasks in a curriculum or design new RL algorithms with inductive biases that favor the types of curricula non-expert human teachers use more frequently.

In this work, we task non-expert humans with designing a curriculum for an RL agent and evaluate the different curricula designs they produced. Specifically, in our RL domain, an agent needs to learn to complete different tasks that are specified with textual commands in a variety of simulated home environments using reinforcement and/or punishment feedback. Human participants are told the target environment on which the agent will be tested on, and their goal is to select a sequence of training tasks that will result in the agent learning the target task as quickly as possible. Our results show that 1) most users successfully identified the two most important concepts the agent needed to learn to complete the target task when designing curricula, 2) users tended to gradually introduce more complex concepts to the agent within each curriculum, and even across different curricula, and 3) different users chose to increase task complexity in different ways and it was significantly affected by the ordering of the presentation of the source tasks. We also find some interesting salient patterns followed by most users when selecting tasks into the curriculum, which could be highly useful for the design of new RL algorithms that accommodate human teaching strategies.

2 Background and Related Work

The concept of curriculum learning was proposed by Bengio *et al.* [2009] to solve the non-convex optimization task in machine learning more efficiently. Motivated by their work, considering the case where it is hard to measure the easiness of examples, Kumar *et al.* [2010] developed a self-paced learning algorithm to select a set of easy examples in each iteration, to learn the parameters of latent variable models in machine learning tasks. Similarly, Lee and Grauman [2011] proposed a self-spaced approach to solve the visual category discovery problem by self-selecting easier instances to discover first, and then gradually discovering new models of increasing complexity.

Although previous work has shown that machine learning algorithms can benefit from curriculum strategies [Bengio *et al.*, 2009; Kumar *et al.*, 2010; Lee and Grauman, 2011], there is limited work on curriculum learning in the context of RL. However, there are several areas related to curriculum learning for RL. Wilson *et al.* [2007] explored the problem of multi-task RL, where the agent needed to solve a number of Markov Decision Processes drawn from the same distribution to find the optimal policy. Sutton *et al.* [2007] extended the idea of lifelong learning [Thrun, 1996] to the RL setting, considering the future sequence of tasks the agent could encounter. Both cases assume a sequence of RL tasks is presented to a learner, and the goal is to optimize over all tasks rather than only the target task. The idea of active learning [Cohn *et al.*, 1996] was also exploited in RL domains [Mihalkova and Mooney, 2006; Vigorito and Barto, 2010] to actively maximize the rate at which an agent learns its environment's dynamics.

Of existing RL paradigms, transfer learning [Taylor and Stone, 2009] is the most similar to curriculum learning. The main insight behind transfer learning is that knowledge learned in one or more source tasks can be used to improve learning in one or more related target tasks. However, in most transfer learning methods: 1) the set of source tasks is assumed to be provided, 2) the agent knows nothing about the target tasks when learning source tasks, and 3) the transfer of knowledge is a single-step process and can be applied in different similar domains. In contrast, the goal of curriculum learning is to design a sequence of source tasks for an agent to learn such that it can develop progressively more complex skills and improve performance on a pre-specified target task.

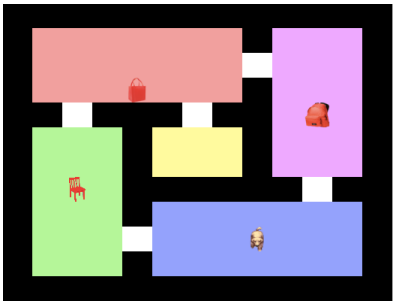

Figure 1: An example five-room layout with one virtual dog, one chair, bag, and backpack with the same color in our domain. It is also the target environment (target command: "move the bag to the yellow room") used in our user study.

Taylor *et al.* [2007] first showed that curricula work in RL via transfer learning by gradually increasing the complexity of tasks. Narvekar *et al.* [2016] developed a number of different methods to automatically generate novel source tasks for a curriculum, and showed that such curricula could be successfully used for transfer learning in multiagent RL domains. However, none of their work explicitly investigates curriculum design from the perspective of human teachers. We think it is natural to consider what humans do when designing curricula since it might be easier for them to capture some examples that are "too easy" (*e.g.*, does not help to improve the current model) or "too hard" (*e.g.*, long training times are needed before the current model could capture this example) for the agent to learn. Such an idea has been studied in the context of teaching humans (*i.e.*, the *zone of proximal development* [Vygotsky, 1978]) but not in agent learning.

3 Problem Formulation

In this section, we first define our sequential RL task with natural language command learning. Then, we introduce a curriculum design problem for non-expert humans.

3.1 Language Learning with Reinforcement and Punishment

Our domain is a simplified simulated home environment of the kind shown in Figure 1. The domain consists of four object classes: agent, room, object, and door. The visual representation of the agent is a virtual dog, since people are familiar with dogs being trained with reinforcement and punishment. The agent can deterministically move one unit north, south, east, or west, and pushes objects by moving into them. The objects are chairs, bags, backpacks, or baskets. Rooms and objects can be red, yellow, green, blue, and purple. Doors

(shown in white in Figure 1) connect two rooms so that the agent can move from one room to another. The possible commands given to the agent include moving to a specified colored room (*e.g.*, "move to the red room") and taking an object with specified shape and color to a colored room (*e.g.*, "move the red bag to the yellow room").

In this sequential domain, the agent needs to learn to respond appropriately to different natural language commands in a variety of simulated home environments using reinforcement and/or punishment feedback. The learning algorithm for this study [MacGlashan *et al.*, 2014] connected the IBM Model 2 (IBM2) language model [Brown *et al.*, 1990] with a factored generative model of tasks, and the goal-directed SABL algorithm [Loftin *et al.*, 2015] for learning from feedback. In SABL, feedback signals from a trainer are modeled as random variables that depend on the policy the trainer wants the agent to follow and the last action the agent took in the previous state. In general, reinforcements under this model are more likely than punishments when the agent selected an action consistent with the desired policy, and *vice versa* for punishment when the action was inconsistent. Using this model of feedback, SABL computes and follows the maximum likelihood estimate of the trainer's target policy given the history of actions taken and the feedback that the trainer has provided. We adapted SABL to this goal-directed setting by assuming that goals are represented by MDP reward functions and that the agent has access to an MDP planning algorithm that computes the optimal policy for any goal-based reward function. In contrast to previous work, we focus on studying how humans perform in designing curricula rather than in training the agent with reinforcement and punishment. Therefore, in this study, the human participants only choose the training curriculum, and the reinforcement and punishment on each of the curriculum's tasks is carried out by an automated trainer, and is observed by participants.

Using this probabilistic trainer model and a curriculum from a human participant, an iterative training regime over each task in the curriculum proceeds as follows. First, the agent receives an English command. From this command, a distribution over the possible tasks for the current state of the environment is inferred using Bayesian inference. This task distribution is used as a prior for the goals in goal-directed SABL. The agent is then trained with SABL for a series of time steps, while the explicit reinforcement and/or punishment feedback is given at random times by the automated trainer. After completing training, a new posterior distribution over tasks is induced and used to update the language model via weakly-supervised learning. After the language model is updated, training begins on the next task and command from the curriculum.

As the agent learns additional tasks, it becomes better at "understanding" the language, successfully interpreting and carrying out novel commands without any reinforcement and punishment. For example, an agent might learn the interpretation of "red" and "chair" from the command "move the red chair," and the interpretation of "blue" and "bag" from the command "bring me the blue bag," thereby allowing correct interpretation of the novel command "bring me the red bag."

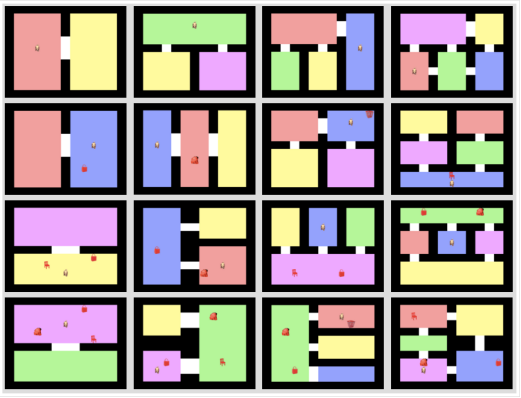

Figure 2: A library of environments provided in a 4×4 grid. They are organized according to the number of rooms and number of objects. There is a command list for each of the 16 environments.

3.2 Curriculum Design

Here, we introduce a curriculum design problem for non-expert humans in our sequential RL domain, where the goal is to design a sequence of source tasks $M_1, M_2, \ldots, M_n$ for an agent to train on such that it can complete the given target task M_t quickly with little explicit feedback. Each source task M_i is defined by a training environment, initial state, and a command to complete in that environment.

To aid our study of how humans form curricula for the agent to train on, we provided subjects a library of environments with different levels of complexities shown in the 4×4 grid in Figure 2. We organized the space of source environments a human could choose to include in their curriculum along two dimensions: the number of rooms and the number of moveable objects present in the environment. The cross product of these factors defines the overall complexity of the learning task, since these factors determine how many possible tasks the agent could execute in the environment and therefore how much feedback an agent could require to identify what the intended task is. For example, the environment in the top left of Figure 2 has the least complexity, because the only possible task the agent can complete is going to the yellow room. In contrast, the bottom right environment has the highest complexity, because the agent could be tasked with going to either the green, red, yellow, or blue rooms; or taking the bag, chair, or backpack to any of the rooms (excluding the room in which the object originates). For the ease of description, we number the environments in the grid from 1 (top left) to 16 (bottom right), from left to right and top to bottom.

After selecting an environment to include in the curriculum, users select the corresponding command to be taught in it from a predefined list of possible commands. For example, the possible commands for environment 5 (second row and first column of Figure 2) are "move to the red room," and "move the bag to the red room."

The target task (shown in Figure 1) has the maximum number of differently colored rooms and shaped objects. In the user study, it is shown on the right side of the grid to remind users the goal of the designed curriculum, but it cannot be

selected as part of the curriculum (enforcing a separation between training and testing).

Note that when we list the possible commands for each environment, we do not include the command that will be used in the target task ("move the bag to the yellow room"). That is, for any environment that contains a bag, the only possible command is "move the bag to the red/green/blue/purple room" even when there is a yellow room. We are interested in studying whether users can figure out that they can construct a curriculum that includes the command "move to the yellow room" and the command "move the bag to the red/green/blue/purple room" to provide the learning agent enough information to master the target command.

We varied the order of the 16 environments in the grid to study the effect of the ordering of source tasks on human performance in designing curricula. Specifically, we transposed the grid, swapping environments 1 and 16, 2 and 12, 3 and 8, *etc.*, such that the difficulty level of the environments gradually decreases from left to right, and top to bottom. Participants were assigned to one of two experimental conditions which varied the ordering of source tasks in the grid:

- **Gradually Complex Condition**: the number of rooms increases from left to right, and the number of objects increases from top to bottom (Figure 2).
- **Gradually Simple Condition**: the number of rooms gradually decreases from top to bottom, and the number of objects gradually decreases from left to right.

4 User Study

To study whether non-expert humans (*i.e.*, workers on Amazon Mechanical Turk, known as "Turkers") can design good curricula for an RL agent, we developed an empirical study in which participants were asked to select a sequence of source tasks for an agent to train on such that it can complete the target task quickly with little explicit feedback.

In our user study, human participants must first pass a color blind test before starting the experiment since the training task requires the ability to identify different colored objects. Second, participants fill out a background survey indicating their age, gender, education, history with dog ownership, dog training experience, and the dog-training techniques they are familiar with. Third, participants are taken through a tutorial that 1) walks them through two examples of the dog being trained to help them understand how the dog learns to complete a novel command successfully using reinforcement and punishment feedback, and 2) teaches them how to design and evaluate a curriculum for the dog. Participants are told that 1) their goal is to design a sequence of source tasks the dog will train on such that the dog can successfully complete the given target task quickly, and 2) higher payment would be given to the Turker if the dog performs well in the target task.

Following the tutorial, participants are requested to select environments and corresponding commands in any order to design their own curricula. Recall that the target task is shown on the right side of the screen to remind participants of the goal for the designed curricula. Upon finishing designing a curriculum (containing at least one task), participants can choose to evaluate their curriculum, watching the automatic trainer teach the agent the entire curriculum. Then, participants are required to redesign the curriculum at least once. We ask participants to explain their strategy for designing the initial curriculum and what things they identified that the dog needed to learn in the curriculum to successfully complete the target task. Participants were also required to explain how they redesigned the curriculum. Participants had the option of providing any additional comments about the experiment.

5 Results

This section summarizes the results of our user-study, which was run on Amazon Mechanical Turk (AMT). We consider data from 80 unique workers, after excluding 17 responses which we identified as users who simply pushed through the AMT task as fast as possible to be paid. We identified such users as those whose completion time was shorter than 5 minutes (the average completion time was 15 minutes 43 seconds, with a standard deviation of 8.8 minutes) or if both designed curricula contained only a single task. There were 40 participants for each of the experimental conditions (gradually complex and gradually simple).

5.1 Participant Performance

Recall that participants were told that their goal was to design a curriculum the dog would train on such that the dog could successfully complete the novel command "move the bag to the yellow room" in the target environment (Figure 1) with little explicit feedback. Therefore, we first examined whether users could successfully identify the need to communicate color and object concepts separately in their curriculum in two experimental conditions. We measured this by analyzing the percentage of users who included both the command regarding moving to any colored room and the command contained any move-able object. Results in Table 1 (the last row) show that in the gradually complex condition, 60% of users captured the idea of teaching the agent both color and object references separately in their initial curriculum, and this number increased to 75% in their final curriculum. The gradually simple condition showed exactly the same results.

Then, we were interested in studying whether users could figure out to teach the agent two more specific concepts separately—the yellow room (the room the agent needs to move to in the target task) and the bag object (the object the agent needs to move in the target task). We evaluated this by computing the percentage of users who combined the command "move to the yellow room" and the command "move the bag to the red/blue/green/purple room" in their curriculum. Surprisingly, in the gradually complex condition, only 23% of users introduced the yellow room and bag concept to the agent in their initial curriculum, and 17% more users captured this idea in their final curriculum. The gradually simple condition produced similar results. However, there is still some evidence showing that more users tended to teach the agent these two specific concepts the agent needed to learn in the target task. Specifically, in the gradually complex condition, we find that 1) 78% of users tried to train the agent to move to the yellow room, and 2) a total of 65% of participants wanted to teach the agent to move an object

Table 1: Summary of percentage of participants for different command selections in two experimental conditions

#	Selected Command	Gradually Complex Con		Gradually Simple Con	
		Initial Cur	Final Cur	Initial Cur	Final Cur
1	move to the yellow room	78%	75%	58%	55%
2	move to the yellow/red/blue/green/purple room	95%	90%	85%	85%
3	move the bag to the red/blue/green/purple room	35%	55%	43%	63%
4	move the bag/basket/backpack/chair to ... room	65%	85%	75%	90%
5	# 1 + # 3	23%	40%	20%	33%
6	# 2 + # 4	60%	75%	60%	75%

(bag/basket/backpack/chair) to some colored room, where 53.8% of them focused on teaching it to move the bag.

In both the initial and final curricula, a chi-squared test shows that the number of users who selected each type of commands in Table 1 was not significantly different ($p > 0.05$) between the two experimental conditions, suggesting that the ordering of source environments does not affect human performance in identifying the concepts the agent needs to learn to complete the target task.

5.2 Concept Introduction

We hypothesized that users would gradually introduce more complex environments or commands to the agent in their curriculum. To validate this, we analyzed the changes in the environment and command complexity. We found that in the gradually complex condition, only 37.5% (or 45%) of users consistently increased environment complexity in their initial (or final) curriculum. However, a total of 50% (or 60%) of users selected the simple command regarding moving to some colored room first, and then consistently chose more complex object-moving commands in their initial (or final) curriculum. The gradually simple condition showed similar results. This suggests that users preferred to consistently introduce more complex commands rather than environments to the agent in each curriculum. A chi-squared test shows that the number of users who consistently introduced more complex environments or commands was not significantly different ($p > 0.05$) between two experimental conditions.

There is another interesting finding that users tended to introduce more complex commands to the agent across different curricula in both experimental conditions. In particular, in the gradually complex condition, for the 37 users who kept or increased the curriculum length, 54% of them only replaced the command regarding moving to some colored room with more complex object-moving command, or added new object-moving commands in the final curriculum. In the gradually simple condition, 62% of the 34 users who kept or increased the curriculum length only introduced more complex object-moving commands to the agent in their final curriculum. Therefore, as we expected, both within a curriculum and between curricula, users tended to gradually introduce more complex commands to the agent rather than more complex environments.

5.3 Transition Dynamics

Although previous results show that less than half of users consistently increased the environment complexity in their curriculum, we observed that a considerable number of users implemented this in segments. It suggests that most users considered increasing the environment complexity when designing curricula. A better understanding of how users select more complex environments might give us insights into the active selection of better curricula.

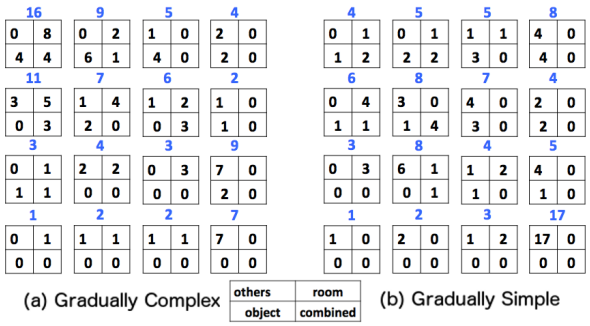

Figure 3: The number of times each of four transitions being followed for each environment in the initial curricula in the two experimental conditions. There are 16 corresponding squares for 16 environments. The blue number represents the total number of times all four transitions being followed in each environment.

We hypothesized that different users would choose to increase the environment complexity in different ways, and it might be affected by the ordering of source environments. In particular, for the 4×4 grid (shown in Figure 2), we defined four different ways for users to increase the environment complexity: room transition, object transition, combined transition, and others. For a given task M_i in a curriculum, a transition to M_{i+1} is a *room transition* if and only if the number of rooms increases between M_i and M_{i+1}. If the number of objects increases, it is an *object transition*, and if they both increase it is a *combined transition*. All other cases are considered as *other transitions*. We aim to study the most popular transition followed by users in two experimental conditions by computing the frequency of each of four transitions being followed for each environment.

Figure 3 summarizes the number of times each transition type (room, object, combined, and other) was used from each environment in the initial curricula in the two experimental conditions. We observe that the room transition was the most-frequently used in the gradually complex condition, while the object transition was the most-frequently used in the gradually simple condition. A chi-squared test shows that

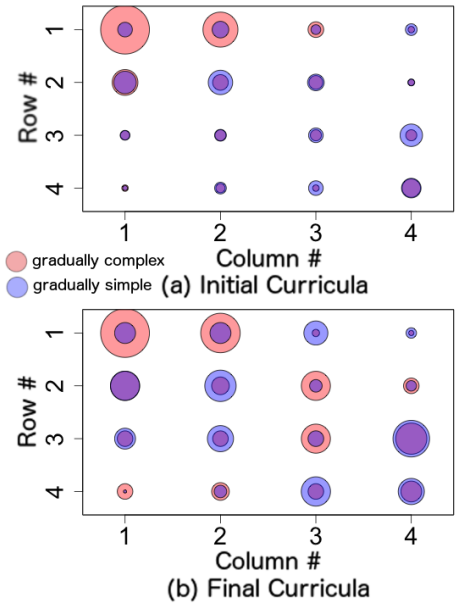

Figure 4: The probability of each environment being included in the initial or final curricula in the two experimental conditions. The purple circle represents the overlap of probability.

the differences of the total number of times each transition type being followed when users design initial curriculum between two experimental conditions was statistically significant ($p \ll 0.01$), verifying that the ordering of source environments does affect the way humans use to increase the environment complexity.

5.4 Environment Preference

We hypothesized that some source environments in the grid would be preferred by users when designing their curricula. Analyzing the properties of these environments might enrich the general principles regarding efficient curricula and inspire the development of new machine learning algorithms that accommodate human teaching strategies. Therefore, we explored user preference in each environment by computing the ratio of the number of users who selected corresponding environment at least once to the total number of users.

Figure 4 summarizes user preference in each of the 16 environments when designing an initial or final curricula in two experimental conditions. A larger dot represents a higher probability of the corresponding environment being chosen. We find that when designing initial curriculum, users were more likely to select 1) Environments 1, 2, 5, and 16 in the gradually complex condition, and 2) Environments 5, 6, 12, and 16 in the gradually simple condition. This finding implies that users preferred to choose 1) the simplest environments that only contain one important concept (Environments 1 and 2 are the two simplest ones that refer to a yellow room, and Environment 5 and 6 are the two simplest ones that include an object) that the agent needed to learn for the target task, and 2) more complex environments that are more similar to the target environment (Environment 12 and 16 are two of the most similar ones to the target environment).

Compared to the initial curricula, Figure 4 shows that most environments had a higher probability of being included in the final curricula in the two experimental conditions, due to the fact that most users tended to increase the curriculum length. In particular, Environments 7, 11, and 12, and 3, 10, 12, and 15 gained the most probability in the gradually complex and gradually simple conditions, respectively. As discussed before, users tended to focus on teaching the agent more complex object-moving tasks (building on previous tasks) when redesigning curricula, and most of these environments provide a good chance for the agent to learn object reference with a relatively large number of different colored rooms.

We also note that users had a lower probability of choosing the two simplest environments (1 and 2) after varying the order of the 16 environments. Fisher's exact test shows that the frequency of each of the 16 environments being selected by users into initial or final curricula was not significantly different ($p > 0.05$) between the two experimental conditions, suggesting that the ordering of source environments does not influence participants' preference in choosing environments. We believe that knowing users prefer 1) isolating complexity, 2) selecting simplest environments they can to introduce one complexity at a time, 3) choosing environments that are most similar to the target environment, and 4) introducing complexity building on previous tasks rather than backtracking to introduce a new type of complexity can be highly useful for the design of new machine learning algorithms which accommodate human teaching strategies.

6 Conclusions and Future Work

In this paper we present an empirical study designed to explicitly explore how non-expert humans design curricula for an agent to train on, allowing the agent to complete a target task with little explicit feedback. Our most important finding is that users followed some salient patterns when selecting and sequencing environments in the curricula, which we plan to leverage in the design RL algorithms in the future. Our goal will be to develop inductive biases in learning algorithms that can benefit from the types of tasks and transitions non-expert human teachers use more frequently.

Future work will 1) allow users to create a sequence of novel source tasks for the agent to train on, 2) come up with a stable way to show the score of the designed curricula to motivate users to design better ones, and 3) implement an RL algorithm that can leverage all interesting salient patterns followed by non-expert humans to design better curricula.

Acknowledgements

This research has taken place in part at the Intelligent Robot Learning (IRL) Lab, Washington State University and the CIIGAR Lab at North Carolina State University. IRL research is supported in part by grants AFRL FA8750-14-1-0069, AFRL FA8750-14-1-0070, NSF IIS-1149917, NSF IIS-1319412, USDA 2014-67021-22174, and a Google Research Award. CIIGAR research is supported in part by NSF grant IIS-1319305.

References

[Bengio *et al.*, 2009] Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. Curriculum learning. In *Proceedings of the 26th annual international conference on machine learning*, pages 41–48. ACM, 2009.

[Brown *et al.*, 1990] Peter F Brown, John Cocke, Stephen A Della Pietra, Vincent J Della Pietra, Fredrick Jelinek, John D Lafferty, Robert L Mercer, and Paul S Roossin. A statistical approach to machine translation. *Computational linguistics*, 16(2):79–85, 1990.

[Cohn *et al.*, 1996] David A Cohn, Zoubin Ghahramani, and Michael I Jordan. Active learning with statistical models. *Journal of artificial intelligence research*, 1996.

[Griffith *et al.*, 2013] Shane Griffith, Kaushik Subramanian, Jonathan Scholz, Charles Isbell, and Andrea L Thomaz. Policy shaping: Integrating human feedback with reinforcement learning. In *Advances in Neural Information Processing Systems*, pages 2625–2633, 2013.

[Knox and Stone, 2009] W. Bradley Knox and Peter Stone. Interactively shaping agents via human reinforcement: The tamer framework. In *The Fifth International Conference on Knowledge Capture*, September 2009.

[Kumar *et al.*, 2010] M Pawan Kumar, Benjamin Packer, and Daphne Koller. Self-paced learning for latent variable models. In *Advances in Neural Information Processing Systems*, pages 1189–1197, 2010.

[Lee and Grauman, 2011] Yong Jae Lee and Kristen Grauman. Learning the easy things first: Self-paced visual category discovery. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pages 1721–1728. IEEE, 2011.

[Loftin *et al.*, 2015] Robert Loftin, Bei Peng, James MacGlashan, Michael L Littman, Matthew E Taylor, Jeff Huang, and David L Roberts. Learning behaviors via human-delivered discrete feedback: modeling implicit feedback strategies to speed up learning. *Journal of Autonomous Agents and Multi-Agent Systems*, pages 1–30, 2015.

[MacGlashan *et al.*, 2014] J. MacGlashan, M. L. Littman, R. Loftin, B. Peng, D. L. Roberts, and M. E. Taylor. Training an agent to ground commands with reward and punishment. In *Proceedings of the AAAI Machine Learning for Interactive Systems Workshop*, 2014.

[Mihalkova and Mooney, 2006] Lilyana Mihalkova and Raymond J Mooney. Using active relocation to aid reinforcement learning. In *FLAIRS Conference*, pages 580–585, 2006.

[Narvekar *et al.*, 2016] Sanmit Narvekar, Jivko Sinapov, Matteo Leonetti, and Peter Stone. Source task creation for curriculum learning. In *Proceedings of the 15th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2016)*, Singapore, May 2016.

[Skinner, 1958] Burrhus F Skinner. Reinforcement today. *American Psychologist*, 13(3):94, 1958.

[Sutton and Barto, 1998] Richard S Sutton and Andrew G Barto. *Introduction to reinforcement learning*, volume 135. MIT Press Cambridge, 1998.

[Sutton *et al.*, 2007] Richard S Sutton, Anna Koop, and David Silver. On the role of tracking in stationary environments. In *Proceedings of the 24th international conference on Machine learning*, pages 871–878. ACM, 2007.

[Taylor and Stone, 2009] Matthew E Taylor and Peter Stone. Transfer learning for reinforcement learning domains: A survey. *The Journal of Machine Learning Research*, 10:1633–1685, 2009.

[Taylor *et al.*, 2007] Matthew E. Taylor, Peter Stone, and Yaxin Liu. Transfer Learning via Inter-Task Mappings for Temporal Difference Learning. *Journal of Machine Learning Research*, 8(1):2125–2167, 2007.

[Taylor, 2009] Matthew E. Taylor. Assisting Transfer-Enabled Machine Learning Algorithms: Leveraging Human Knowledge for Curriculum Design. In *The AAAI 2009 Spring Symposium on Agents that Learn from Human Teachers*, March 2009.

[Thomaz and Breazeal, 2006] Andrea Lockerd Thomaz and Cynthia Breazeal. Reinforcement learning with human teachers: Evidence of feedback and guidance with implications for learning performance. In *AAAI*, volume 6, pages 1000–1005, 2006.

[Thrun, 1996] Sebastian Thrun. Is learning the n-th thing any easier than learning the first. In *Advances in Neural Information Processing Systems*, volume 8, pages 640–646, 1996.

[Vigorito and Barto, 2010] Christopher M Vigorito and Andrew G Barto. Intrinsically motivated hierarchical skill learning in structured environments. *Autonomous Mental Development, IEEE Transactions on*, 2(2):132–143, 2010.

[Vygotsky, 1978] L. S. Vygotsky. *Mind in Society: Development of Higher Psychological Processes*. Harvard University Press, 1978.

[Wilson *et al.*, 2007] Aaron Wilson, Alan Fern, Soumya Ray, and Prasad Tadepalli. Multi-task reinforcement learning: a hierarchical bayesian approach. In *Proceedings of the 24th international conference on Machine learning*, pages 1015–1022. ACM, 2007.