Gaining Expertise through Task Re-Representation

Connor Wilhelm and Dan Ventura
Computer Science Department
Brigham Young University
Provo, UT 84602 USA
connor.wilhelm@byu.edu, ventura@cs.byu.edu

Abstract

In the field of computational creativity, machine learning is becoming a popular choice for modeling (domain- or task-specific) expertise. Unfortunately, such modeling is often very expensive when performed on the naturalistic representation of a task, which can be information sparse and thus ineffective for creative reasoning over the domain/task. We propose *task distillation* as a mechanism for *re-representation* of machine learning tasks as small datasets that contain all the information needed to gain expertise in the task, resulting in two key outcomes: the ability to efficiently teach acquired expertise and an explicit "cognitive" artifact that can be used for task understanding, potentially facilitating creative discovery. We demonstrate task distillation on two reinforcement learning problems: cart-pole and Atari *Centipede*, reducing them to single-batch datasets that can be learned by new agents in a single learning step and argue this re-representation therefore demonstrates the "essence" of the task/domain.

Introduction

The representation of a task determines how we perceive and interact with the task. Roman numerals, while fully capable of representing numbers, make multiplication and division difficult; while Indo-Arabic numerals make simple operations more intuitive. The decimal system's spread from India was vital to Al Khwarizmi's development of intuitive algorithms and algebra. Fibonacci's use of the system was vital for his own creative advances to mathematics, and led the way for Newton and Leibniz to develop calculus (Dasgupta, Papadimitriou, and Vazirani 2006). Good representations of a task are required for productive creative output, and good representations are not always natural or obvious. Thus, re-representation is often a key step in the creative process.

Re-representation is also a vital part of teaching. Al-Khwarizmi explained algebra through complex geometric word problems, yet the modern representation of algebra as alphanumeric equations is simple enough to teach children.

We present *task distillation* as a computational model of re-representation designed for teaching. This generalization of dataset distillation (Wang et al. 2018) involves transforming a given learning task into a smaller, more quickly learned synthetic task that can be used to train a model such that the model's performance approximates the performance of a model trained directly on the original task. The synthetic task is a highly compressed representation that is more information-dense than the original task's representation. Most machine learning tasks rely on naturalistic data representations that sample from some real-world data distribution, while the synthetic task is free from naturalistic constraints and can be significantly reduced in size. To provide a concrete example of task distillation, we distill reinforcement learning environments into single-batch synthetic supervised learning datasets that can be learned in a single step of stochastic gradient descent (SGD). We show that the task distillation meta-learning process creates a new representation, the synthetic task, capable of being used to teach the task to a variety of learners. We do not claim that learners shown in our simple examples develop creative solutions during this process, but we argue that they do gain the task-specific expertise that is a precursor to potential creativity. In addition, this new representation is a compact cognitive artifact that can aid in understanding the original task. This small, information-dense representation may be easier to manipulate than the original representation in searching for creative solutions.

Re-Representation

Re-representation is a process by which the features of a given task or object are transformed from their direct representation into another form. The target representation is useful if it can be manipulated into creative solutions in a more obvious way than the original representation (Wiggins and Sanjekdar 2019). This is related to transformational creativity in Boden's theory of creativity: re-representation transforms the creative space of the original representation, such that more intuitive exploration for creative artifacts can be performed (Boden 1990).

Re-representation can be a purely internal process: transforming stimuli to match a representation in memory held in the brain. This is analogous to "seeing as", interpreting a novel stimulus as a familiar object, which can be mentally manipulated (Olteţeanu 2015). However, re-representation can be realized externally, by manipulating physical or meta-physical objects. For example, a sculptor must physically deform stone with a chisel to realize their internally-represented vision of the sculpture. While this final physical representation is the creative artifact, not all re-representations must be creative artifacts; but even re-representation-as-pure-intrinsic-mental-state might be intentionally externalized as an artifact.

Such a re-representation of a task can be utilised to teach basic competency in the task and/or to facilitate further creative problem solving; and this re-representation can be in an entirely different domain than the task itself. The Atari game *Centipede*, for example, requires no natural language skills (Atari 1980), yet expertise in *Centipede* can be taught primarily through natural language. *The Video Master's Guide to Centipede* is one such example, with over 100 pages of natural language and diagrams outlining complex strategies for maximizing score. The guide provides such teaching without a single screenshot of the actual game (Dubren 1982). The author re-represented expertise in playing *Centipede* into natural language, and the reader must re-represent the language back into *Centipede* gameplay.

While teaching expertise can be a creative task in itself, expertise is required for any type of creative task. Expertise can be vital to intentional and efficient searches of a creative space, but it is even more necessary in proving an artifact is creative. While novelty and value are core determiners of creativity, it is the field of the domain which judges novelty and value. Without being able to demonstrate expertise to the field, an otherwise creative artifact will have no impact on the field and be forgotten (Csikszentmihalyi 1996). Expertise can be held by an individual or be distributed throughout a system, but creativity cannot occur without expertise (Reilly 2008).

Task Distillation

In task distillation, one learning task is re-represented as a separate synthetic task that can be used to teach expertise quicker than direct learning on the original task. This is a generalization of dataset distillation, extended to allow for machine learning tasks beyond supervised learning datasets (Wang et al. 2018). In order to demonstrate task distillation, we distill the cart-pole and Atari *Centipede* environments into single-batch supervised datasets. We provide a brief formal definition of task distillation, and then provide examples of its ability to teach through re-representation.

Task distillation consists of producing a synthetic task T_d from a target task T_0, such that T_d contains the compressed teaching potential of T_0 for a distribution of learner models. That is, learners trained on T_d should approximate the performance on an evaluation task of learners trained directly on T_0. In addition, the distilled task should be a compressed representation of the original task: $|T_d| \ll |T_0|$. Thus, T_d can be used instead of T_0 for training to reduce training costs without a significant drop in performance.

Task distillation is not limited to simply compressing a task into a denser representation of the knowledge required to teach. Rather, it can be used to transform a learning task into a different modality. We demonstrate one form of transmodal task distillation by distilling reinforcement learning (RL) environments into synthetic supervised learning (SL) datasets. As a consequence of re-representing an RL environment as an SL dataset, new learners will not need to explore an environment to learn the task once the distilled dataset is created. In our examples, the learners can achieve expertise on the original task by training on the distilled dataset in a single step of stochastic gradient descent with mean squared error loss—significantly cheaper than the reinforcement learning process it replaces.

Methods

We provide experiments that distill two environments: cart-pole and Atari *Centipede*. First, we informally describe our algorithm for generalized task distillation, though other dataset distillation algorithms could also be generalized to this end. Second, we provide implementation details used in our experiments.

Our method for task distillation is based on the meta-learning method for dataset distillation (Wang et al. 2018). This method utilizes a nested loop: the inner loop trains a new learner on the distilled task, and the outer loop uses the trained learners' performance on the real task to update the distiller to teach the learners to better perform on the task. We provide a diagram (Figure 1) to show the meta-learning process for distilling an arbitrary task into a synthetic task. A formalization of the algorithms for task distillation and RL-to-SL distillation is beyond the scope of this work.

In each experiment, we distill a reinforcement learning environment for a set of learners with the same architecture. We utilize proximal policy optimization (PPO) as the outer loss function, and include an auxiliary critic network that is optimized alongside the distiller on the PPO loss. The critic is required for calculating PPO policy loss and is discarded when training is completed (Schulman et al. 2017). The architectures and hyperparameters are standard for direct-learning PPO on cart-pole and Atari, respectively. [1]

We create our distiller by parameterizing a randomly-initialized dataset of the dimensions we want for our final dataset. Each instance must match the dimensions of the target environment's state space in order to fit in the learner networks. The number of instances in the final distilled dataset is a hyperparameter and can only be optimized through experimentation. The synthetic data instances are updated directly by the optimization algorithm. Soft labels are used and optimized (Sucholutsky and Schonlau 2021b), being represented as a vector matching the size of the environment's action space. Other formulations are possible, such as the generative teaching network (GTN). This formulation involves training a generator network to produce the distilled data, allowing for more than one dataset to be produced. The GTN is capable of distilling cart-pole (Such et al. 2020); however, training the distilled set directly appears to be more effective, especially for Atari environments. In addition, we preferred a single high-quality re-representation, while the GTN generates many varied re-representations.

For cart-pole, distillation success was determined by

[1] See the following blog for standard PPO implementation details: `https://iclr-blog-track.github.io/2022/03/25/ppo-implementation-details/`. We utilized all these details except for learning rate annealing and value loss clipping for both our direct RL and distillation experiments.

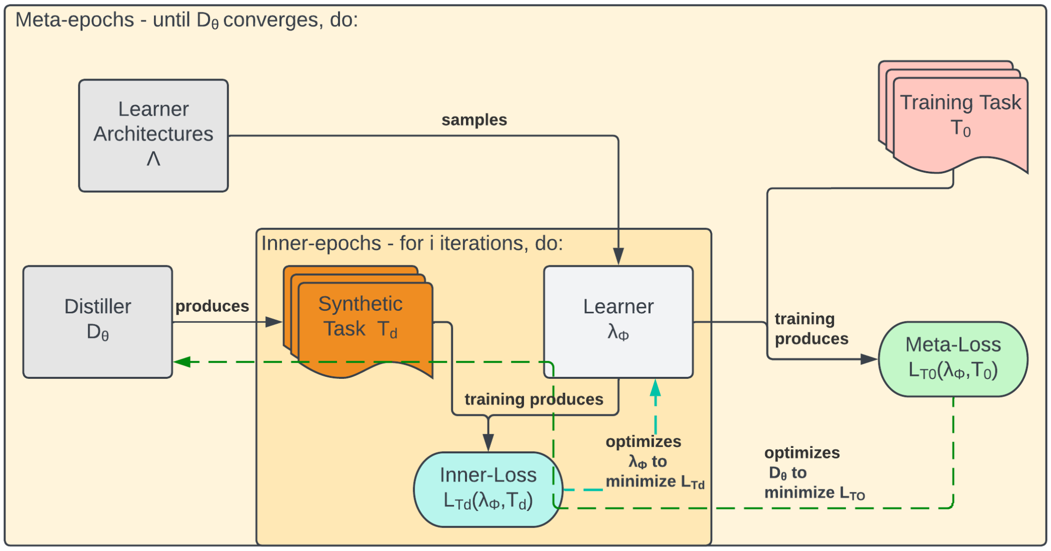
Meta-epochs - until D_θ converges, do:
Learner Architectures Λ
samples
Training Task T_0
Inner-epochs - for i iterations, do:
Distiller D_θ
produces
Synthetic Task T_d
Learner λ_Φ
training produces
Meta-Loss $L_{T0}(\lambda_\Phi, T_0)$
training produces
Inner-Loss $L_{Td}(\lambda_\Phi, T_d)$
optimizes λ_Φ to minimize L_{Td}
optimizes D_θ to minimize L_{TO}

Figure 1: The meta-learning process for generalized task distillation. The inner loop involves training a newly sampled learner on the synthetic task. The trained learner is tested against the real task, and the loss is backpropagated through the inner learning process back into the distiller. This repeats until the distiller converges, which can be seen by the learners' average performance.

whether the distilled data could teach randomly sampled models from the learner set to fully solve the task (defined somewhat arbitrarily as attaining a reward of 500). For *Centipede*, which does not have a "solved" state (but rather is a more open-ended problem of score maximization), we compare the average reward reached by distiller-trained models to the reward reached by a PPO agent. This is reasonable because distillation's meta-learning relies on the same loss function as direct learning and thus has the same limitations.

Cart-pole Distillation

As a standard toy problem for reinforcement learning, cart-pole demonstrates the advantages of distillation and shows how distillation re-represents the entire environment as a small dataset that can be used to understand the cart-pole task. The cart-pole problem is a simple classical control problem based on a physical system. A cart is connected to a pole by a hinge. The pole begins nearly perfectly upright. The agent must move the cart either left or right each timestep to attempt to keep the pole balanced atop the cart. If the pole rotates past a certain threshold in either direction, or if the cart moves past a threshold, the simulation ends. The goal is to balance the pole as long as possible, up to a maximum of 500 timesteps.

This task is easily solved by deep reinforcement learning. We distill this problem into a single-batch representation for supervised learning using randomly initialized agents with the same architecture. The resulting distillation set can be used to train all models sampled from the learning set to balance the pole up to the time limit, solving cart-pole.

It took approximately 3.5 times the number of cart-pole episodes for the distillation to converge compared to direct RL learning. With the increased overhead of meta-learning, distillation was approximately 6 times slower. However, the end result of distillation is a single 2-instance dataset that can teach the cart-pole task in one SGD step (see Figure 2 for a visualization). Thus, distillation can be a cheaper alternative to sequentially training 6 or more RL agents. In addition, the distilled dataset is an artifact that can be used to more easily interpret the original task, simplifying cart-pole's infinite state space into two key examples.

We have experimented with a variety of distilled dataset sizes and have determined that all dataset sizes, above a certain threshold, are capable of being used to solve the problem. The minimum sized teaching set is most interesting, as it is the densest learning representation possible using distillation. Interestingly, this also appears to be more human-interpretable, providing an explainability artifact that shows the "essence" the task. As shown in Figure 2, cart-pole's continuous state space is distilled into two discrete examples that completely characterize the task: showing the pole leaning left in one and right in the other. Neither the state transition function nor the reward function are directly modeled; instead, the action labels clearly demonstrate that to maximize reward the cart must simply be moved in the direction the pole is leaning. While this does not explain the whole model of the system's physics, it shows how to move the cart to balance the pole, which is all that is needed to solve the task. In addition to its explainability potential, the 2-example distilled dataset is also the cheapest learning representation, though negligibly so compared to other one-batch distilled datasets.

For cart-pole, this minimum teaching dataset contains only two instances. This is the theoretical limit for environ-

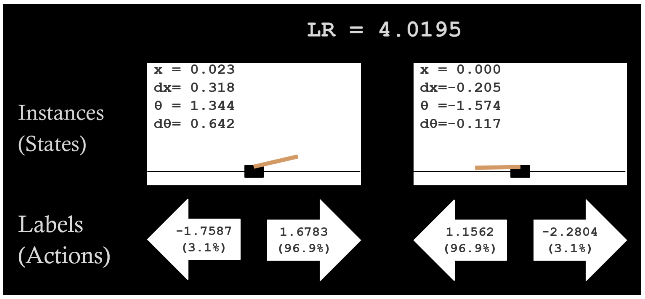

Figure 2: The minimum-sized distilled teaching set for the cart-pole environment. Training on this set for a single step of SGD can teach the cart-pole task to any member of the learner set. The state vectors are shown numerically and visually. The action labels are provided as raw values as well as a softmaxed policy for the provided state. Note that the state is not a valid cart-pole state: the environment would have ended after the pole reached $\theta = \pm 0.2095$, and the simulator does not work for values beyond $\theta = \pm 0.418$. This demonstrates that the distilled instances are not copies of data seen during distillation training; they are synthesized.

ments with only two actions that are required for solving the problem; the teaching set must provide a distinction between when to use these two actions (Sucholutsky and Schonlau 2021a). Notice that in the continuous state space of cart-pole, there are virtually infinite possible states, restricted only by the computer's precision. However, as we can see in Figure 2, the strategy for cart-pole can be described in two states: the cart should move left with a left-leaning pole, and right with a right-leaning pole. While this simple strategy does not address edge-cases, such as when the cart is near the edge of the screen, it is still sufficient to solve the problem.

Centipede Distillation

The Atari 2600 environments represent a significant increase in difficulty from cart-pole by greatly increasing the state space dimensionality, the action space, and the complexity of strategies required to perform well on the environment. We demonstrate that complex reinforcement learning environments can be distilled by successfully distilling a teaching dataset of only 10 instances, which can be used to train the learners to perform well on *Centipede*. This is the theoretical minimum sized dataset required to teach *Centipede*; given we are using soft-label vectors and *Centipede* has 18 distinct actions (Sucholutsky and Schonlau 2021a).

Unlike cart-pole, there is no well-defined solution to *Centipede*: a player's goal is to maximize score. Reaching the theoretical maximum score is well beyond the capabilities of small reinforcement learning agents, given our resources. Therefore, we judge distillation success by comparing an individual's cumulative reward on *Centipede* after training on the distilled task versus training on *Centipede* directly.

Direct learning on *Centipede* yields an average reward of 9167 points after approximately $1,000$ epochs of training. Distillation yields an average reward of 8083 points on *Centipede* after approximately $8,000$ epochs of training, reaching 88% of the reward in 8 times the number of optimization steps. The drop in average reward is expected: we are compressing knowledge gained from testing against *Centipede*; it is unlikely that a distillation of a task can be used to teach a learner to perform at a higher level than can the original task itself. However, the learners still perform well above random, and the best-performing learner trained on that distilled data achieved a score of $36,978$—well above the human average of $11,963$ (Mnih et al. 2015).

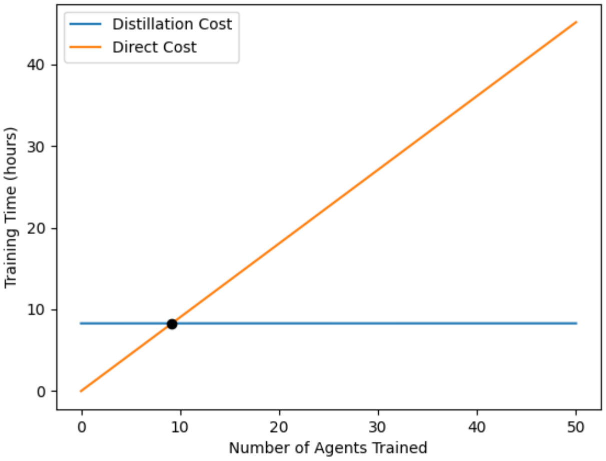

Figure 3: Time costs for training *Centipede* agents using distillation versus direct task learning. While training the distiller is costly, distillation training time increases negligibly (by 0.18 seconds) as the number of agents trained increases. Direct learning is significantly cheaper for a single agent but must be repeated in full for each additional agent; when training more than 9 agents, distillation is cheaper. As the number of agents trained increases, distillation becomes more cost-effective compared to direct learning.

The cost increase is also expected: distillation pays much of the learning cost up-front. Each epoch of direct RL training on *Centipede* using our resources took on average 3.25 seconds, compared to an average of 3.73 seconds per epoch for distillation. While the distillation process takes approximately 9.18 times as long as direct RL training ($8,000$ epochs x 3.73 seconds/epoch vs $1,000$ epochs x 3.25 seconds/epochs), the benefits of training on the distilled set is clear. Training on the distilled data is significantly cheaper than the full direct RL training. It takes on average 0.18 seconds to train a model on the distilled data, 18,000 times faster than RL training. This speedup is due to distillation removing the requirement to interact with the environment, as well as the amount of data trained on: 10 instances for training on the distilled data versus 8,000,000 instances for training on the environment. Using our resources, it is more time-effective to utilize distillation rather than directly learning on *Centipede* if one is training more than 9 models. See Figure 3 for the training time costs of distillation and direct learning on *Centipede* as a function of number of learners trained.

Similar to the distilled cart-pole instances, the distilled *Centipede* instances represent invalid states. Their values go beyond the range of valid pixel values and cannot be accurately represented as images. However, despite not being intuitive representations that resemble real *Centipede* states, these representations are capable of teaching the learners. While this representation is not as easily interpretable as the cart-pole distillation, it re-represents the task to efficiently impart expertise to the learners. Even so, the dataset provides another artifact that can be examined alongside the environment and the agents to more fully explain the learning process on the environment and potentially lead to creative behavior invention.

Discussion

Our experiments demonstrate the re-representation of the cart-pole and *Centipede* learning tasks as compressed representations that teach through a different learning mode. The re-representations do not contain all information about the original environments: there is no indication of the range of states, the state-transition function, or the reward function. Rather, only pertinent learning information is stored.

While the systems described in this work are not creative, the systems contain the expertise which is a necessary prerequisite for creativity (Reilly 2008). The expertise, gained through many iterations of learner training and testing, is aggregated within the distilled task. Upon learning, the expertise is imparted to a learner which can perform the targeted task. If one needed expertise in a creative system, a pre-distilled set is a much quicker alternative to gaining expertise by learning on the whole task. With a dataset distilled from a reinforcement learning environment, a model can be trained in seconds rather than hours of exploration. The resources required to explore the environment's state space to gain expertise can instead be used toward exploring a creative space. In addition, this re-representation provides another way to understand the environment, one which could be manipulated to allow for the invention of creative and interesting strategies in the environment.

For example, consider a simple creative system that utilizes distillation to create a cart-pole agent capable of performing tricks, which could be used in a novel balancing routine. Without distillation, this might be done by providing a variable reward function, which is changed to reinforce policies that lead to interesting and novel behavior, as judged by a separate evaluation function. The space of reward functions can then be searched to find reward functions that result in producing creative behaviors (as judged by the evaluation function). However, without distillation, each point in reward function space can only be tested by a full session of (expensive) reinforcement learning. Distillation can be used instead—the search can be performed directly on the distilled training set's parameter space. Testing a point in this re-represented space can be performed more efficiently than using RL: one inexpensive SGD step and one episode of performance on cart-pole to demonstrate the learned behavior and receive a score from the evaluator model.

A search for creative and interesting strategies in *Centipede* can benefit from distillation in much the same way as cart-pole. Searching through the smaller distillation space, compared to the parameter space of a complex reward function, as well as cheaper training for evaluation, would provide a significant speedup to the creative search. The time saved evaluating each point could then be put toward searching more points in the space, allowing for a more thorough examination of the creative space, and potentially finding a superior creative artifact than could be found using the same resources without distillation.

References

Atari. 1980. Centipede. Arcade, Atari 2600.

Boden, M. 1990. *The Creative Mind*. Abacus.

Csikszentmihalyi, M. 1996. *Creativity*. HarperCollins Publishers, 1st edition. chapter 2, 27–31.

Dasgupta, S.; Papadimitriou, C.; and Vazirani, U. 2006. *Algorithms*. McGraw-Hill Education. chapter 0, 2.

Dubren, R. 1982. *The Video Master's Guide to Centipede*. Bantam Books.

Mnih, V.; Kavukcuoglu, K.; Silver, D.; Rusu, A. A.; Veness, J.; Bellemare, M. G.; Graves, A.; Riedmiller, M.; Fidjeland, A. K.; Ostrovski, G.; Petersen, S.; Beattie, C.; Sadik, A.; Antonoglou, I.; King, H.; Kumaran, D.; Wierstra, D.; Legg, S.; and Hassabis, D. 2015. Human-level control through deep reinforcement learning. *Nature* 518:529–533.

Olteţeanu, A.-M. 2015. "Seeing as" and re-representation: Their relation to insight, creative problem-solving and types of creativity. *Publications of the Institute of Cognitive Science*.

Reilly, R. C. 2008. Is expertise a necessary precondition for creativity?: A case of four novice learning group facilitators. *Thinking Skills and Creativity* 3(1):59–76.

Schulman, J.; Wolski, F.; Dhariwal, P.; Radford, A.; and Klimov, O. 2017. Proximal policy optimization algorithms. *arXiv* 1707.06347.

Such, F. P.; Rawal, A.; Lehman, J.; Stanley, K. O.; and Clune, J. 2020. Generative teaching networks: Accelerating neural architecture search by learning to generate synthetic training data. In *Proceedings of the 37th International Conference on Machine Learning*, 9206–9216.

Sucholutsky, I., and Schonlau, M. 2021a. "Less than one"-shot learning: Learning N classes from M < N samples. In *Proceedings of the 35th AAAI Conference on Artificial Intelligence*, volume 35, 9739–9746.

Sucholutsky, I., and Schonlau, M. 2021b. Soft-label dataset distillation and text dataset distillation. In *Proceedings of the International Joint Conference on Neural Networks*, 2220–2227.

Wang, T.; Zhu, J.-Y.; Torralba, A.; and Efros, A. 2018. Dataset distillation. *arXiv* 1811.10959.

Wiggins, G. A., and Sanjekdar, A. 2019. Learning and consolidation as re-representation: Revising the meaning of memory. *Frontiers in Psychology* 10.