SESiL: Social, Evolutionary Supported Learning

Tianshu Zhao
Carleton University
Ottawa, Canada
Tianshu.Zhao@Carleton.ca

Zinovi Rabinovich
Carleton University
Ottawa, Canada
Zinovi.Rabinovich@Carleton.ca

ABSTRACT

Social Learning describes several variations of interaction between a learning agent and a source of (potentially) beneficial behaviour. Mainly, though outliers exist, there are three forms of this interaction. First is the obvious “monkey-see-monkey-do”, learning by the imitation of an observed behaviour. Second, the “teacher-learner” relationship, where an experienced agent actively guides or instructs the learner. Finally, knowledge extraction from observation, where an agent generalizes from the observed interactions of others with the environment to its own needs and goals. However, in spite of the enormous volume of work in social learning, once commonality persists — an agent can directly benefit from the additional information and improve its own behaviour. But what happens if the agent has identified the benefits of other’s behaviour, but cannot absorb them?

In this paper, we study the support that social learning can garner from an evolutionary perspective on the process: rather than absorbing additional behavioural information directly, agents share and merge their behavioural information by choosing a mate. It is the children that represent and carry the socially learned, combined behaviour. We term the combined learning process SESiL (Social, Evolutionary Supported Learning). Besides the formal definition of the framework, we provide experimental studies of its properties. Specifically, we deploy SESiL in multi-tasking classification. Starting from a population of agents who have been partially-pretrained on small subsets of labels, we give them the agency to seek and choose a mate based on the observed classification performance. Presuming availability of a “genetic merger” operator (in our case, classifier network merger), we allow the mutually-agreed mating pairs to be replaced by two children that carry their (imperfectly) merged knowledge. We baseline SESiL against a full-data access classifier, a distributed learner (split-learn-merge) and several forms of more classical evolutionary compute, where agents have no say in choosing a mate, but are bred following their overall performance.

KEYWORDS

Multi-Agent Learning; Social Learning; Evolutionary Computing

ACM Reference Format:
Tianshu Zhao and Zinovi Rabinovich. 2026. SESiL: Social, Evolutionary Supported Learning. In *Proc. of the 25th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2026), Paphos, Cyprus, May 25 – 29, 2026*, IFAAMAS, 9 pages. https://doi.org/10.65109/LPWP5477

Proc. of the 25th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2026), C. Amato, L. Dennis, V. Mascardi, J. Thangarajah (eds.), May 25 – 29, 2026, Paphos, Cyprus. © 2026 International Foundation for Autonomous Agents and Multiagent Systems (www.ifaamas.org). https://doi.org/10.65109/LPWP5477

1 INTRODUCTION

Social Learning (SL), the ability to extract useful behavioural patterns from observation of others, is a much needed [11], long coming [18], and now slowly growing, discussion topic in the global field of Artificial Intelligence, and Multi-Agent Systems in particular [3, 4, 10, 30].

Social Learning naturally links with such paradigms as federated/distributed learning [20, 21, 37], multi-task learning[25, 29, 33], and meta-learning[34, 41, 42]. These paradigms disregard the social aspect, creating rigid relationships between different agents. However, they do pursue one common purpose: merge multiple sources of behavioural information into one generalised behavioural principle. SL distributes these ideas and introduces limited observability of the behavioural information source.

The renewed interest notwithstanding, the underlying principles of SL have another name: Evolutionary Computing (EC), where a solution is found by continual recombination and refinement of solution candidates based on a general fitness to perform a task (or a set of them) [14, 32]. It was, therefore, quite inevitable that SL and EC will, pun somewhat unintended, recombine. In fact, one recombination is quite well studied: Culture Evolution, with, perhaps, the most recent work in this area being the “Cultural Accumulation in Reinforcement Learning” [8]. Now, nearly as a rule, in combination with SL or stand-alone, evolutionary computing would be used as a tool, rather than the subject of investigation from the AI perspective. One very notable exception to this is the work by Livnat [28] that studies the computational benefits of sexual reproduction. Interestingly, Livnat’s work suggests that recombination and, even, genome mutation are affected by low-level strategic choices by genes. Though Livnat disagrees with Dawkins [9] on the selfishness of those strategies. More interestingly though, for our discussion, is the parallel of these strategic choices to Social Learning Strategies [23], which we do not see as accidental. In fact, we’d like to suggest, hopefully quite provocatively, that **Evolution** (based on sexual reproduction) **is a Social Learning Algorithm**. The central idea can be summarized as follows: “I observe beneficial patterns in others and wish to internalize them, yet I cannot incorporate them directly into myself—so I pass them on to my descendants”. Notably, this process extends beyond simple imitation; it is not merely “monkey-see-monkey-do”—for the other monkey must also consent to the exchange. As with other evolutionary methods, our method also begins from a population of agents that specialize in different tasks. However, instead of selecting parents solely based on performance (i.e., allowing duplication and unconditional mating), our method grants agents the right to make and accept/reject mating proposals, based on an individual-centric mate selection metric (details in Section 3.3). Parenting, therefore, requires a bi-directional, mutual proposal acceptance.

In this paper, we formalise the above idea and present SESiL, a novel, evolutionary supported, social learning framework based on agency in mate selection and black-box recombination/mutation operators. We explore the features of SESiL by applying it to a social multi-task learning scenario based on multi-class image classification. In particular, as a basic sanity check, we show that SESiL does not result in a significant performance loss compared to a full-data-access, centralised classifier. We also compare SESiL with the baseline of non-evolutionary break-and-merge approaches, i.e. training individual models on sub-sets of classes and merging the resulting models. We then investigate how quickly the population can adopt new knowledge. First, we introduce an “outlander” model, pre-trained on additional image classes, previously unseen by the population. We then compare the performance of the resulting population with that of a full-data-access, centralised classifier fine-tuned on the new classes. We also investigate the performance of fine-tuning the population on additional image classes, rather than an "outlander" injection. Finally, we run an ablation study of the mating strategy by gradually removing the agency from mate-choice. It shows that SESiL’s distributed, bi-directional agreement-based mate-choice is pivotal to its success.

2 RELATED WORKS

2.1 Social Learning

Bandura [1] has developed a Social Learning Theory, where he argues that humans learn new behaviour from more than just personal experience. He argued for a social component, learning by observing the experience of others, and focusing on cognitive and information processing abilities of humans. It is this focus, we believe, that makes Social Learning an ideal tool for artificial intelligence.

Of course, observations of others can take the form of social signals/rewards. E.g., Isbell et al. [18] experiment with social learning in a reinforcement learning agent, Cobot, inside the LambdaMOO online chat community. Cobot adapts its behaviour based on the cues and reward signals provided by human participants of LambdaMOO. Since Isbell’s paper, Reinforcement Learning with Human Feedback (RLHF) became a flourishing feild of its own [7, 22, 24].

A key component of social learning is, of course, to know from who, what and when to learn. In fact, it is best to make such choices strategically. Kendal et al. [23] review the concept of social learning strategies (SLSs), highlighting the need to update it, rathen than adopting a fixed strategy. This follows from the evidence that individuals flexibly switch and combine strategies. SLS remain a valuable framework for linking psychology, neuroscience, and evolutionary biology. In fact, it has been argued that social learning (and its strategy) are an evolutionary phenomenon, be that cultural evolution [6, 31] or the classic one [17, 38].

2.2 Multi-Agent Learning with Social Dynamics

Borsa et al. [4] demonstrate that observational learning can emerge naturally from learning, showing that an agent—without explicitly modeling others—can learn by observing another agent’s effects on the environment and adjusting its behavior accordingly. Ndousse et al. [30] study an emergent form of social learning. By constraining training environments and adding a model-based auxiliary loss, agents in [30] develop generalized social learning policies that enable them to acquire complex skills, adapt to novel environments, and even outperform solo-trained agents. Bhoopchand et al. [3] propose a method for few-shot real-time imitation that enables agents to exhibit cultural transmission, learning new behaviors from humans in novel contexts without pre-collected data, thus laying groundwork for cultural evolution in artificial intelligence. Derstroff et al. [10] introduce a peer learning framework where multiple agents learn together by exchanging action advice, model teacher selection as a multi-armed bandit, and show that this collaborative setup outperforms single-agent and baseline methods on challenging OpenAI Gym tasks.

2.3 Evolutionary Computing

Feng et al. [14] present the first comprehensive and systematic introduction to evolutionary multi-task (EMT) optimization, detailing the application of EMT algorithms to a wide range of optimization problems. Cook et al. [8] present cultural accumulation, i.e., building knowledge and skills across generations through both in-context and in-weights learning, leading to better performance than single-lifetime training. Of course, we cannot forego the mention of the recently released book on NeuroEvolution [32], that overviews several decades of research into evolutionary strategies of learning behaviours encoded by neural networks.

Taking a more Lamarckian view, evolution may occur in the Knowledge Space, leading to Cultural Evolution (CE). Reynolds et al. [31] present CE algorithms as a foundational framework for modelling human social interaction. Their work covers CE’s theoretical origins, core structure, and evaluation. In CE, cultural knowledge is seen as a nearly-common, shared attribute of a society, and it is in an arms race with its users. Users seek to expand, modulate, and update current culture with new experiences, while the knowledge continues to reaggregate and drive the behaviour of individuals. Notably, Bourahla et al. [5] show that good CE performance can be achieved with reduced elitism (where only the best performers transmit their knowledge to others), without initialisation noise (perfect information transmission from parents), and complex adoption of teachers via a social proto-network. Alas, even this advanced work presumes that individuals have no agency in mate selection. For our SESiL approach, this agency is pivotal.

Hence, of key importance to our paper are the works by Livnat [27] and Werner [39]. The former argues that mutation, "while not Lamarckian, or ‘directed’ to increase fitness", is not random. We adopt a similar approach in SESiL, where mutation is accumulated over time, until an individual becomes more attractive as a mate. The latter paper, by Werner, is the only one we could find that actually considers a consensual mating based on behavioural success. Werner studies the evolution of communication, and only those individuals that communicate their intent successfully to each other mate. We take this idea to its ultimate form: mating is a bi-directional choice, based on mutual estimates of mating benefits.

2.4 Merging of Neural Network

Now, Neuroevolution [32] considers a multitude of genetic "crossover" functions that mix-and-match and *merge* parental capabilities. In this paper, though SESiL framework does not depend on

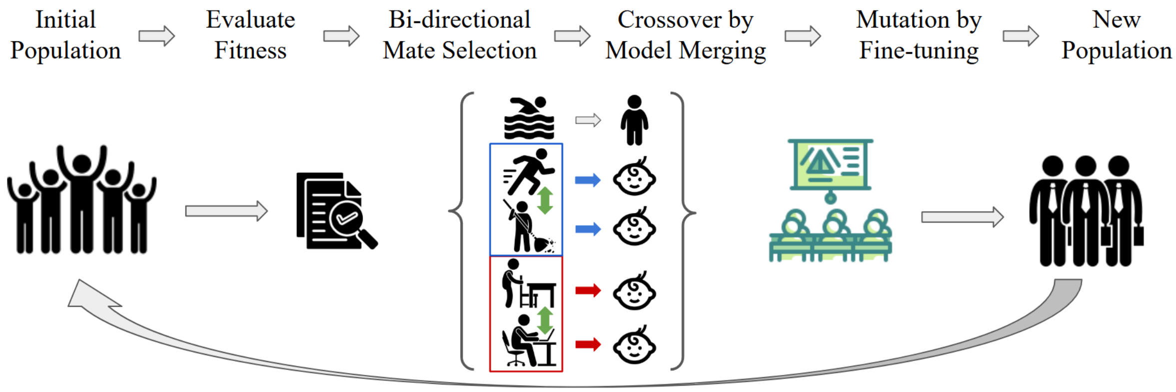


Figure 1: The evolutionary structure of our method differs from the standard genetic algorithm.

a particular cross-over operation, our experiments are based on a neural-merger process. This is quite common for fine-tuning and expanding LLMs (e.g., [36] and references therein).

Now, merging can be both naive and a highly non-trivial operation. Garipov et al. [15] discover that the optima of deep neural network loss landscapes are connected by high-accuracy paths, and based on this insight, propose Fast Geometric Ensembling (FGE), a method that efficiently trains high-performing model ensembles in the time of a single model, outperforming prior ensemble techniques. Entezari et al. [12] show that accounting for neural networks' permutation invariance leads to barrier-free paths between stochastic gradient descent (SGD) solutions, support this with empirical evidence and preliminary theory, and discuss implications for the lottery ticket hypothesis, distributed training, and ensembles. Izmailov et al. [19] propose Stochastic Weight Averaging (SWA) – averaging multiple points along an SGD trajectory – which finds flatter solutions, improves generalization across various neural networks, approximates FGE with a single model, and requires minimal computational overhead. Stoica et al. [35] propose ZipIt!, a method to merge two models of the same architecture trained on different tasks into a single multi-task model without retraining, using a feature-wise "zip" operation and partial merging to create multi-head models, where needed.

2.5 Distributed and Federated Learning (DFL)

Though we are yet to incorporate this form of merger, followed by specialisation, into SESiL, we find it necessary to acknowledge DFLs for future work. Esmaeili et al. [13] propose holonic learning (HoL), a collaborative and privacy-preserving framework that structures distributed deep learning into self-similar hierarchical "holons" for flexible model aggregation and communication. Goel et al. [16] propose SocialLight, a scalable distributed cooperation learning method that uses locally centralized critics and counterfactual reasoning to estimate each agent's contribution. Li et al. [26] introduce a distributed training framework with parallel curriculum experience replay that collects diverse experiences and automatically assesses subtask difficulty. Jiang et al. [21] propose federated heterogeneous policy distillation (FedHPD), a method for black-box federated reinforcement learning that enables knowledge sharing among heterogeneous agents via action distributions. Jacopo et al. [20] propose a highly compressed federated learning framework that represents network weights via a smaller trainable parameter vector multiplied by a fixed sparse matrix, drastically reducing communication costs while maintaining accuracy, and provide theoretical insights linking training-by-sampling to random convex geometry. Tang et al. [37] propose a fuzzy clustered federated learning method that partitions clients' data to generate personalized models, effectively reducing heterogeneity.

3 SOCIAL, EVOLUTIONARY SUPPORTED LEARNING

In this section, we describe the evolutionary components of our method. We first present the details of the mating metric, which defines how an agent scores and selects other agents to form pairs. We then describe the genetic crossover process, through which beneficial traits from partners are incorporated into their descendants. In our experiments, beneficial traits are defined as a model's performance on different tasks, namely its skills. Individuals in the population are designed such that benefits are treated either as improvements to existing skills or as expansions of their task domain through the acquisition of new ones.

3.1 Overview of Evolutionary Structure

Our evolutionary method (see Figure 1 for the overall structure) builds a sequence of populations (generations) $\{\mathcal{P}_t \mid 0 \leq t \leq T\}$, where T denotes the number of generations and each generation $\mathcal{P}_t$ consists of N multi-tasking neural networks (to we will refer interchangeably as agents, individuals, or models). Both for the ease of discussion, as well as in our experiments, we will presume that individuals are multi-class (multi-task) classifiers. Models of generation zero, $\mathcal{P}_0$, are presumed to be partially pre-trained on a small subset k of labels from the task domain. In this way, models can be viewed as possessing "skills" that may overlap (identical), partially overlap (joint), or not overlap (disjoint) across tasks. To build the next generation, we begin by evaluating the per-class accuracies of each model, which serve as the model's fitness and provide the basis for subsequent mate selection. We do not directly perform crossover between the optimal pairs with the highest accuracies. Instead, we introduce a mating metric that allows each individual model to maintain its own "view" of what is valuable in a mate. Based on this, we employ a probabilistic bidirectional mate selection (see Section 3.3), in which a mating pair forms only if the mate selection is reciprocated (e.g., model A selects B, and model B selects A). For the genetic merger of a pair ("cross-over"), we utilize network merging approaches, such as ZipIt! [35], treated as black-box operators. Once the merged models (i.e., offsprings) are obtained, we slightly fine-tune them as a form of mutation, after which the offspring join the population for the next generation, replacing the parents. Unmated models *may* persist as well.

3.2 Fitness Evaluation

Given population $\mathcal{P}_t$, we evaluate the accuracy of its models on every class in the domain of tasks, obtaining an accuracy vector $\mathbf{f}^i$, where f^i_j denotes the accuracy of the i'th model in $\mathcal{P}_t$ on the j'th class.

The set of fitness vectors for the entire population $\mathcal{P}_t$ is denoted by $\mathcal{F}_t = \{\mathbf{f}^i \mid 0 \leq i < N_t\}\}$. Note that, in general, population size N_t can change from generation to generation (see, e.g., Section 3.3.3), but in most cases we keep it fixed. Each model is trained only on a randomly selected subset of k classes. We refer to classification on these k classes as the "skills" of the model. However, it is the full fitness vector, $\mathcal{F}_t$, that is used for mate selection.

3.3 Mating Metric

Given two pre-trained models, A and B, from the population, each model can be regarded as a specialist trained on k randomly selected classes. In the mating function, when model A looks at model B: (i) a hyper-parameter threshold, τ, is used to determine whether a model **"knows" a given class**, e.g., if accuracy $> \tau = 0.5$. (ii) the classes that both models already know are treated as common skills; (iii) the classes unknown to A but for which B achieves good performance are considered extra skills that contribute to diversity. Thus, the "attractiveness" score of model B from model A's perspective depends on how well B complements A's skill set. We use the three factors shown above to compute an attractiveness score of one model to another.

3.3.1 Mate scoring. Let us define $S(A \rightarrow B)$, an attractiveness of model B as a mate from model A's perspective. If model A has accuracy vector $\mathbf{f}^A$ and model B has accuracy vector $\mathbf{f}^B$, we define

$$K_A = \{j \mid f^A_j > \tau\}, \quad K_B = \{j \mid f^B_j > \tau\},$$

as the sets of known classes for models A and B, respectively, where f^A_j (or f^B_j) denotes the accuracy of the j-th class. Then the extra skills of B relative to A can be represent by $E_{B|A} = K_B \setminus K_A$, and the common skills are given by $C_{A,B} = K_A \cap K_B$. The mating score of model B as evaluated by model A is defined as

$$S(A \rightarrow B) \;=\; w_{\text{ext}} \sum_{j \in E_{B|A}} f^B_j \;+\; w_{\text{com}} \sum_{j \in C_{A,B}} f^B_j, \tag{1}$$

where τ is the threshold introduced in (i), and the relative weights w_{ext} and w_{com} are two more hyper-parameters of the score.

3.3.2 Probabilistic bidirectional mate selection. Let $\mathbf{Z}_t = [z_{i,j}] \in \mathbb{R}^{N_t \times (N_t - 1)}$ denote the score matrix at generation t, where $z_{i,j} \geq 0$ is the score assigned by individual i to individual j ($i \neq j$).

Step 1: Probabilistic mate choice. Each individual i selects a candidate mate $j \neq i$ with probability

$$P_t(i \rightarrow j) = \begin{cases} \dfrac{z_{i,j}}{\sum_{k \neq i} z_{i,k}}, & \text{if } \sum_{k \neq i} z_{i,k} > 0, \\ \dfrac{1}{N_t - 1}, & \text{otherwise.} \end{cases} \tag{2}$$

Step 2: Bidirectional acceptance. A pair (i, j) is accepted into the mating set $\mathcal{M}_t$ if and only if the choice is reciprocated:

$$(i, j) \in \mathcal{M}_t \iff \text{mate}(i) = j \text{ and } \text{mate}(j) = i,$$

where $\text{mate}(i)$ is sampled according to $P_t(i \rightarrow j)$.

Step 3: Quota completion. If $|\mathcal{M}_t| < N_t$, repeat step 1 and step 2 until $|\mathcal{M}_t| = N_t$. We impose a maximum number of repeats to limit the number of mating attempts for each model. For each successful pair, crossover is performed twice, producing two offspring that replace their parents in the population. For models without a mate, we allow them to survive into the next generation.

3.3.3 Population Shrinking. As an analytical and baseline tool, we use a mate selection variant where unmated models do not survive. As a result, the number of successful matings $|\mathcal{M}_t|$ satisfies $|\mathcal{M}_t| \leq N_t$ and $N_{t+1} = |\mathcal{M}_t|$. Consequently, as the population becomes less diverse over successive generations, the number of reciprocated choices decreases and eventually converges to one.

3.4 Black-box for the Genetic Crossover

Crossover [2, 32] is a process in which two parent solutions exchange parts of their "genetic" information to produce new offspring that combines traits from both. In general, SESiL does not dictate the genetic cross-over method – it is a black-box. However, in our experiments, the genetic crossover operator between mated models is a neural-merger. As a kind of ablation study in support of the SESiL framework, we experiment with three merger methods.

Model merging builds on mode connectivity [15], where models fine-tuned from the same initialization can be combined through simple weight interpolation [19, 40]. The first merging method we consider, *weight-average merging*, is based on this idea:

$$W^*_i = \gamma W^A_i + (1 - \gamma) W^B_i. \tag{3}$$

However, independently trained models often lie in different modes and contain misaligned neurons, which can severely degrade performance [12]. *Permutation-based merging* addresses this issue by reordering hidden units prior to interpolation:

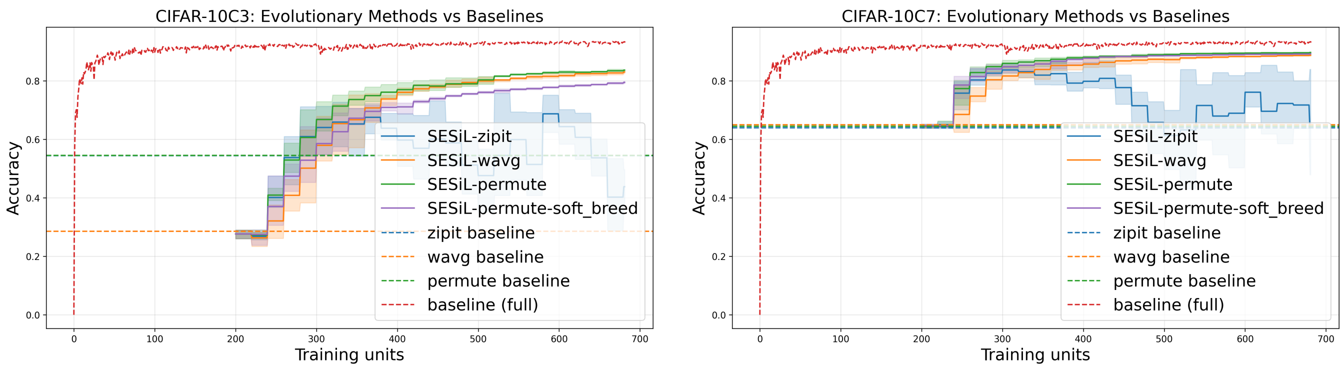
$$W^*_i = \gamma W^A_i + (1 - \gamma) P_i W^B_i P^T_{i-1}, \tag{4}$$

where γ is the coefficient, W is the weight parameters of a model, and P is the permutation matrix.

However, permutation-based merging assumes that models can be aligned into the same loss basin, which may not hold when they are trained on different tasks. *ZipIt!* [35] instead aligns intermediate features by learning linear transformations (e.g., $f_i = W_i x + b_i$), and then merges parameters through mapped projections (Eq. 5)

$$W^*_i = M^A_i W^A_i U^A_{i-1} + M^B_i W^B_i U^B_{i-1} \tag{5}$$

Here, M denotes the merge matrix that combines feature pairs into a single shared output space, and U is the corresponding unmerge matrix that reverses this operation.



(a) Comparison on CIFAR-10 dataset with $k = 3$.

(b) Comparison on CIFAR-10 dataset with $k = 7$.

Figure 2: SESiL vs baselines on CIFAR-10. Performance of merge-only (ablation) baselines is marked by horizontal dashed lines.

3.5 Mutation through Fine-tuning

Before adding them to the next generation, the offspring (i.e., merged models of mated pairs), as well as the unmated survivors, undergo a low-degree fine-tuning. This serves as a form of mutation for the newly-born and simulates lifelong learning of the unmated agents. Using a fraction of the pre-training budget of generation zero, the network weights, π^i_θ, are (up)trained. Previously unmated agents are up-trained on the set of classes already known to them, while offspring agents are up-trained on the set of all classes known to their parents. By design, these mutations increase the chances of the unmated agents to find a mate in the next generation. However, in our experiments, they also stabilised the merged, offspring networks.

In real-world multi-task settings, it is generally possible to continue training on previous tasks (at least separately for different datasets, as in DFLs). This approach differs from combining all task data and training a single model from scratch, which is often infeasible when tasks are heterogeneous.

4 EXPERIMENTS

In this section, we evaluate our method's performance. We first show that SESiL achieves performance comparable to training a single, large model from scratch, i.e., standard learning (Section 4.1). We then demonstrate that SESiL exhibits better adaptability in adopting new skills (Section 4.2) and achieves greater efficiency than alternative, baseline evolutionary methods (Section 4.3).

Baselines. As a common baseline, we build a single, large multi-task classifier model, simultaneously trained on all classes and the full (training) data set. We use CIFAR-10 and CIFAR-100 for our experiments. To align the training budget, the baseline model is trained for a total number of iterations equal to the sum total of those used to train all models in the initial population, plus the fine-tuning budget used for mutating the population through the generations. We note that this large model is an upper bound on the expected performance, as SESiL uses imperfect merger of existing, low-grade agents with minimal extra, individual (re)training. Moreover, in certain multi-task settings, training a model on all tasks simultaneously is often infeasible. The lower bound on our expected performance is formed by *pure merging* baselines (using methods in Section 3.4), where all pre-trained models of generation zero are merged at once, followed by fine-tuning on the entire task domain using the same training budget it takes SESiL to produce at least one model that knows all classes.

Hyper-parameters of Mate Selection. In every generation, mating proposals are attempted until $|\mathcal{M}_t| = N$ or the number of attempts reaches the bound of 100.

For the scoring function, the **known class** threshold is set to $\tau = 0.5$, while the weight balance between extra and common skills is set, respectively, to $w_{\text{ext}} = 0.9$ and $w_{\text{com}} = 0.1$.

Crossover and mutation. For crossover, we use ZipIt! [35], weight averaging, or permutation as “black-box” operators. To mutate, each merged model is trained for $1/10$ of the pre-training budget of a model on the subset of combined classes obtained, using a batch size of 128 and a learning rate of 0.001.

4.1 SESiL vs Standard Learning

We consider a population of $N = 10$ or 20 pre-train models with a CIFAR dataset. The full task domain has 10 or 100 classes and we assign each model in the population a subset of size $k \in [3, 7]$ for CIFAR-10, $k \in [10, 20]$ for CIFAR-100 to evaluate the conceptual performance of our method. We adopt the mating setup described in Section 3.3.2, maintaining a fixed population size. We experiment with different merging methods, including weight averaging and permutation. The results demonstrate that differences in performance are largely driven by the mating process itself, while the choice of merging method has only minor impact. Furthermore, SESiL's long-term performance approaches the upper bound of the single, large model baseline.

4.1.1 CIFAR-10. Figures 2a and 2b show the results for $k = 3$ and $k = 7$ under different merging methods for crossover, alongside training the common baseline (a single, large model) from scratch. The evolutionary process begins with an initial population, where each individual possesses only a subset of all skills, i.e., *knows* only a few specific classes. As individuals mate, thus merging their skill sets, their descendants eventually possess good performance across the entire task domain and know all classes. We fine-tune the models slightly as mutation and include the corresponding mutation budget in the graph over generations. We also compare against an ablated SESiL variant – soft-breeding – which we discuss in detail in Section 4.3.1.

The shaded area represents the minimum–maximum accuracy within the population. Starting from the training budget used for the initial population (i.e., 200 units in total, with 20 epochs per model), models with complementary skill sets begin to mate, leading to a rapid increase in overall performance during the first five generations (more details in section 4.3.2). The improvement slows once all models in the population "know" all the skills/classes for the task domain. Subsequent evolution gradually pushes the performance beyond the pure-merging baselines and toward the common training baseline.

Although SESiL is structurally independent of the specific network merge method, the influence of the merger's performance is notable. E.g., ZipIt! [35] looses accuracy when merging two networks with common skill-sets. This is especially notable in later generations in Figures 2a and 2b.

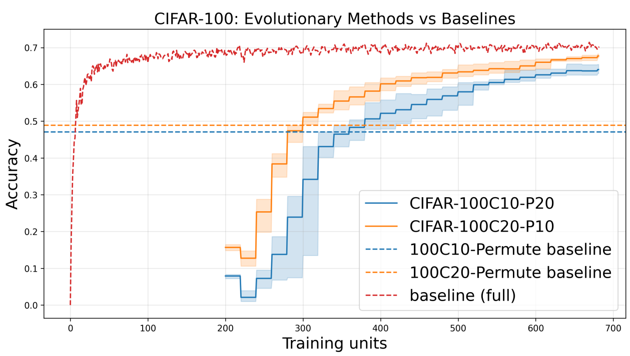


Figure 3: SESiL performance with baselines on CIFAR-100. In CIFAR-100C<k>-P<N_0>, population had <N_0> models, each pre-trained on a subset of <k> classes.

4.1.2 CIFAR-100. We also evaluate our method with larger capacity to cover tasks up to 100 classes (See figure 3). Specifically, we experiment with (i) a "naive" population of 20 models trained on randomly selected $k = 10$ classes out of 100, and (ii) an "elite" population of 10 models trained on randomly selected $k = 20$ classes out of 100. We observe that the population, where the initial models are more knowledgeable (ii), initially achieves better performance than the one with less knowledgeable initial models (i), even though (i) maintains twice the population size and greater diversity. However, in the long-term, the naive population shows very close asymptotic performance. This demonstrates a natural trade-off in social learning between the size of the population and the initial proficiency of individuals within.

Figure 3 shows the performance curves of SESiL in these two scenarios. Initially, individual knowledge very quickly proliferates through the population, leading to a dramatic increase the the performance. However, once every individual's skill set covers the entire task domain (models *know* all classes), improvement rate slows down. Mutation continues to introduce skill improvements, which quickly spread through the population, but this is a slower process overall. Which leads us to our next experiment.

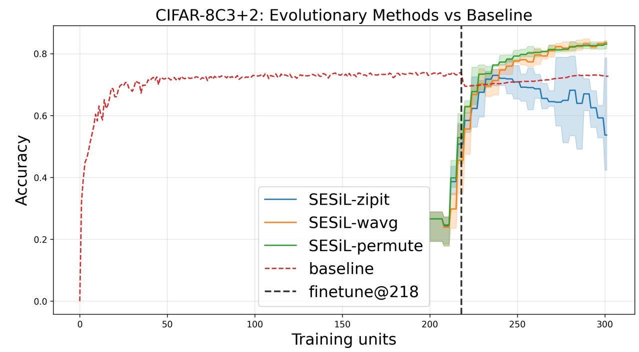


Figure 4: SESiL before and after the arrival of the outlander.

4.2 Adopting New Skills

Previous experiment has shown that SESiL performance approaches its upper bound. But its analysis suggested heightened ability to absorb new knowledge. In this experiment, we test the extent of this ability. We retrain our common baseline on 8 "core" classes and subsequently fine-tune it on the remaining 2. For SESiL, we train $N = 10$ models on 3 randomly selected classes out of the "core" 8 as the initial population. At the fine-tuning stage, we introduce an "outlander" model that has been pre-trained on the 2 remainder classes. The outlander replaces one of the models in the population, maintaining the overall training budget of the original population. The mutation budget in this part of experiments considers all ancestors of the best-performing model in the final generation.

Figure 4 illustrates the performance both before and after the arrival of the outlander, in comparison with fine-tuning the baseline model on the same set of 2 classes. Evaluating on all classes of CIFAR-10, the fine-tuning of baseline model leads to reduced performance on the previously known classes, the overall performance initially decreases but gradually recovers over time. In contrast, SESiL quickly (re)combines skill sets in the population, enabling adoption to the new skills without incurring such a trade-off. Our method achieves superior performance when adopting new tasks. The reason we introduce the outlander and apply fine-tuning at the 5-th generation is that, by this point, at least one model in the population has acquired knowledge of the entire "core" task domain (i.e., all 8 classes), and the performance tends to converge. Note that, models trained on a smaller task domain generally perform better on the corresponding tasks compared to models trained on a larger domain.

We also examine the parameter level of the models on all 8 classes, created by both our method and standard training. After 218 training units as the budget, we select the best-performing models produced by our method with different merging strategies, along with the baseline model, each covering the task domain of 8 classes. We use exact the same fine-tune setting to train 20 units

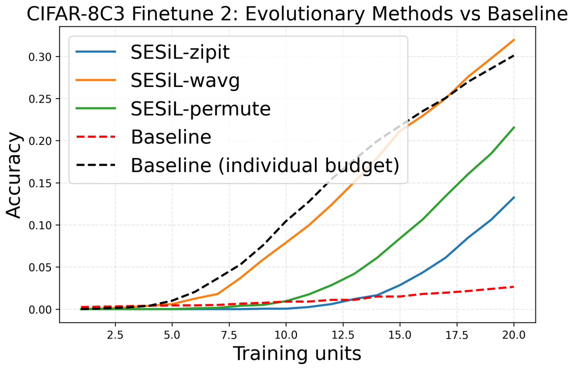


Figure 5: Finetuning SESiL and standard training models.

on only the 2 new classes as well as evaluating their performance. Considering the over-training issue of the baseline, we also apply the same fine-tuning procedure to a baseline trained only with the individual budget (i.e., we use 40 units, but depending on the ancestors). In Figure 5, we observe that the models produced by our method learn faster than or similar to the baseline models from standard training. Surprisingly, the variant of our method with simpler merging strategy gain better performance (see the difference of merging strategies in section 3.4). Note that a baseline model trained with only 40 units of budget is unlikely to outperform our method.

4.3 Ablation Studies

In this section, we present ablation studies of SESiL from two perspectives: mating strategy (Section 4.3.1) and task domain coverage (Section 4.3.2). An additional variation on mutation is also considered in Section 4.3.3.

4.3.1 Mating Strategy Ablation. Here, we attempt to replace SESiL's mating principle by **breeding** strategies, defined as follows.

Let $\mathcal{P}_t = \{a_i\}_{i=1}^N$ denote the population at generation t, where each agent a_i is associated with a fitness value $f(a_i)$. We define a fitness-proportional selection distribution $p_f(a_i) = \frac{f(a_i)}{\sum_{k=1}^N f(a_k)}$, and let $S(a_i \rightarrow a_j)$ denote the (asymmetric) scoring function used by agent a_i to evaluate a_j as a potential mate. In **soft-breeding**, the first parent is sampled uniformly at random from $\mathcal{P}_t$, and the second parent is selected by maximizing the scoring function, i.e., $a_j = \arg\max_{a_k \neq a_i} S(a_i \rightarrow a_k)$. In **guided-breeding**, the first parent is sampled according to p_f, while the second parent is again selected via the scoring function. In **hard-breeding**, both parents are independently sampled from p_f with the constraint $i \neq j$.

We first experimented with soft-breeding over CIFAR-10. As Figures 2a and 2b show, our mating approach extends the individual skill sets faster and yields higher performance overall. Moreover, as the plots' min-max performance shading shows, bi-directional mating selection preserves population diversity longer than breeding.

To obtained a more fine-grain results, we turn to CIFAR-100. Figure 6 includes the comparison of mating and breeding variants. Once the population, or the skills of most models, covers the entire task domain, the performance of our mating approach becomes significantly better than that of the breeding. Interestingly, the hard-breeding variant, which considers only overall accuracy, achieves the best performance among the three breeding variants. Since soft-breeding and guided-breeding also incorporate individual scoring, we observe that bidirectional acceptance plays an important role in how scoring influences pairing. It results in a more complete form of social learning. To coin phrase: "not just monkey-see-monkey-do, because the other monkey has to agree to it too".

Notice that moving from soft-breeding to hard-breeding slowly removes from the mating selection the ability to absorb rare skills from individuals who, overall, do not perform very well. Thus, in breeding processes, the population relies more heavily on the mutation to introduce novelty into elite performers, while SESiL also exploits unique features of overall weak agents.

As mentioned, for both parent selection approaches (mating and breeding), our evaluation uses the same scoring function and merging strategies. This makes the breeding vs mating choice the only factor that accounts for potential performance differences.

As we've noted, the cross-over operations are responsible for the initial fast(er) learning rate of the population. It is, therefore, the efficiency of the mating choice that is key to good evolutionary learning strategy. Our results (figure 6) clearly show that agency in mate choice is more efficient than forced, breeding-styled mating. Although it is possible to simulate the same consideration at the breeding level (after all, matchmaking is a vast industry), breeding practice is hard to implement in a distributed (social) learning scenario. We, therefore, put forward that agency in mating is a major factor in the success of SESiL and any other learning framework with evolutionary support that will follow.

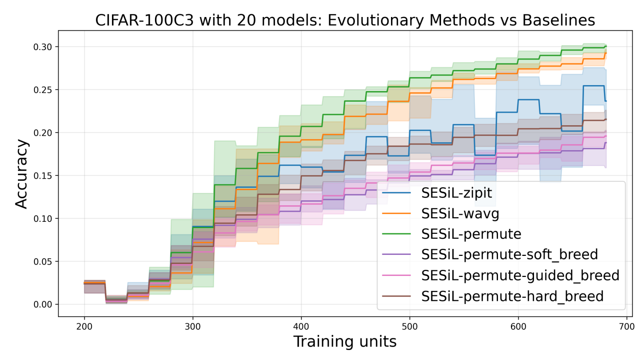


Figure 6: SESiL variants, $k = 3$, population of 20 models.

We assess diversity through the standard deviation of the centralized performance measure within a population. In contrast to Figures 2a and 2b, Figure 7 shows that our mating approach regulates the change in diversity: the initial population exhibits high diversity, which gradually converges over generations. At the same time, overall performance of individuals consistently improves. Now, in general, highly diversified population can much better absorb environmental shocks, which in our experiments would be represented by the introduction of new image classes. However, as we have seen, SESiL maintains a population of flexible agents who continue to absorb new information effectively. We can only marvel at the social equity built by SESiL, with skills nearly universally available and supported by all members of the population.

4.3.2 Domain-coverage by Population. To complete the background on SESiL's performance, we study variations that influence "coverage", i.e., the difference between the total sum of all skills found in the population vs the total number of skills possible.

"Perfect" complementation and population shrinking. First, we address the case where every agent in generation zero has a perfect "complement". We achieve this by producing generation zero elements in pairs. In CIFAR-10, we randomly split the 10 classes into a "5+5" pair, then for each element of the pair we pre-train a model that specialises in corresponding 5 classes, and then add both models to the initial population. As is expected from the mating's preference on extension, the perfect pairs quickly match up, leading to a much faster skill convergence compared to fully randomised sub-selection of skills for the initial population (see Table 1).

We also experiment with the variant in Section 3.3.3, where unmated agents do not survive, and the evolution ends when no mating pair can form. We observe that, across generations, the population size decreases while the task domain covered by the models consistently increases. In our experiments, the final remaining model always encompasses the task domain of the population. Naturally, populations with models that have greater commonality of skill, or compliment each other, converge faster.

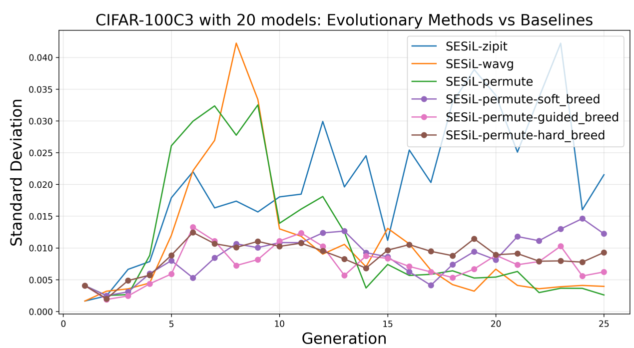


Figure 7: Standard deviations of the centralized performance.

Table 1: Number of generations required for at least one model in the population to acquire all skills.

	10C3	10C7	100C10	100C20	5+5
Generation	4	3	7	6	3

Uncover-able Task Domain. In the previous section, we assumed that the skills of individual models in the population could collectively cover the entire task domain, particularly for the "elite" in such a small "society". Moreover, the mutation in Section 3.5 only affects performance on the known classes. Therefore, we run an experiment to train a population of 20 models on a randomly selected subset of $k = 3$ classes out of 100, which means at most 60 classes are known to the population. We piggyback this experiment on the comparison with breeding variants. As Figure 6 shows, SESiL continues to improve over time, although, in absolute terms, the 100-class accuracy is depressed compared to the full-coverage case.

4.3.3 The extreme mutation. As an ablation study, we introduce an extreme mutation variant in which each model in the population is fine-tuned on only a single, randomly selected class. The results show the performance crashes over just the first few generations. Deeper analysis shows that this extreme mutation triggers *strong forgetting*: marginal performance increase on the new class is accompanied by a significant accuracy reduction on previously known classes. This impacts both negatively impacts offspring stability after parents merge, and the ability of unmated agents to attract mates in the next generation. The latter is due to the fact that mate metric contains a performance threshold that determines whether an agent "knows" a class. If prior knowledge is lost, due to single class fine-tuning mutation effects, then previously "known" classes and, thus, attractive features of an agent will be lost.

This naturally raises the question: what happens if no model performs well on any task? In our approach, mutation is handled by allowing an underperforming model to remain independent, continue training ("exercise"), and then re-enter the mating pool at a later stage. By contrast, breeding in genetic algorithms treats mutation as purely random, which may introduce new skills but often does so inefficiently. Importantly, fine-tuning on entirely new skills is not an effective strategy, as models and humans alike learn more efficiently when focusing on improving existing skills, rather than adopting completely new ones from scratch. While traditional genetic algorithms balance crossover (to preserve strong features) and mutation (to introduce new material), the randomness of mutation generally emphasizes long-term exploration. In contrast, we argue that evolutionary progress can be accelerated in the short term by improving mate selection and designing a more skill-diverse initial population, thereby reducing reliance on random mutation to achieve adaptability.

5 CONCLUSION AND FUTURE WORKS

In this paper, we introduce an evolutionary framework that leverages model merging, treating AI models as agents and encourage them to transfer appropriate skills to their descendants. We propose a mating metric to perform voluntary, bidirectional mate selection in the evolutionary process, creating a new paradigm of social learning: SESiL. We evaluate our method on CIFAR datasets for image classification, and the results show that SESiL, as an evolutionary approach, outperforms standard breeding-based approaches that do not allow agency in mating. In particular, SESiL is far more efficient in exploiting mate selection to promote skill expansion, in contrast to simply relying on mutation to introduce new abilities. Furthermore, our method surpasses standard learning baselines in adopting new tasks and parameter adjustment. In future work, we will expand SESiL applicability, including reinforcement learning variants; introduce meta-learning features into the mating process and take into account domain transfer in the mating score function.

REFERENCES

[1] Albert Bandura. 1977. *Social Learning Theory.* Prentice Hall, Englewood Cliffs, N.J.

[2] Thomas Bartz-Beielstein, Jürgen Branke, Jörn Mehnen, and Olaf Mersmann. 2014. Evolutionary algorithms. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery* 4, 3 (2014), 178–195.

[3] Avishkar Bhoopchand, Bethanie Brownfield, Adrian Collister, et al. 2023. Learning few-shot imitation as cultural transmission. *Nature Communications* 14, 1 (2023), 7536.

[4] Diana Borsa, Nicolas Heess, Bilal Piot, Siqi Liu, Leonard Hasenclever, Rémi Munos, and Olivier Pietquin. 2019. Observational Learning by Reinforcement Learning. In *Proceedings of the 18th International Conference on Autonomous Agents and MultiAgent Systems (AAMAS).* IFAAMAS, Montreal, QC, Canada, 1117–1124.

[5] Yasser Bourahla, Manuel Atencia, and Jérôme Euzenat. 2022. Knowledge transmission and improvement across generations do not need strong selection. In *AAMAS 2022-21st ACM international conference on Autonomous Agents and Multi-Agent Systems.* ACM, 163–171.

[6] Levin Brinkmann, Deniz Gezerli, KV Kleist, Thomas F Müller, Iyad Rahwan, and Niccolò Pescetelli. 2022. Hybrid social learning in human-algorithm cultural transmission. *Philosophical Transactions of the Royal Society A* 380, 2227 (2022), 20200426.

[7] Stephen Casper, Xander Davies, Claudia Shi, Thomas Krendl Gilbert, Jérémy Scheurer, Javier Rando, Rachel Freedman, Tomasz Korbak, David Lindner, Pedro Freire, Tony Tong Wang, Samuel Marks, Charbel-Raphaël Ségerie, Micah Carroll, Andi Peng, Phillip J. K. Christoffersen, Mehul Damani, Stewart Slocum, Usman Anwar, Anand Siththaranjan, Max Nadeau, Eric J. Michaud, Jacob Pfau, Dmitrii Krasheninnikov, Xin Chen, Lauro Langosco, Peter Hase, Erdem Biyik, Anca D. Dragan, David Krueger, Dorsa Sadigh, and Dylan Hadfield-Menell. 2023. Open Problems and Fundamental Limitations of Reinforcement Learning from Human Feedback. *Trans. Mach. Learn. Res.* 2023 (2023). https://openreview.net/forum?id=bx24KpJ4Eb

[8] Jonathan Cook, Chris Lu, Edward Hughes, Joel Z. Leibo, and Jakob N. Foerster. 2024. Artificial Generational Intelligence: Cultural Accumulation in Reinforcement Learning. In *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*, Amir Globersons, Lester Mackey, Danielle Belgrave, Angela Fan, Ulrich Paquet, Jakub M. Tomczak, and Cheng Zhang (Eds.). http://papers.nips.cc/paper_files/paper/2024/hash/6df3a719d99bd2479c04114d357003d0-Abstract-Conference.html

[9] Richard Dawkins. 2016. *The selfish gene.* Oxford university press.

[10] Cedric Derstroff, Mattia Cerrato, Jannis Brugger, Jan Peters, and Stefan Kramer. 2024. Peer Learning: Learning Complex Policies in Groups from Scratch via Action Recommendations. In *AAAI.* 11766–11774.

[11] Edgar A Duéñez-Guzmán, Suzanne Sadedin, Jane X Wang, Kevin R McKee, and Joel Z Leibo. 2023. A social path to human-like artificial intelligence. *Nature Machine Intelligence* 5, 11 (2023), 1181–1188.

[12] Rahim Entezari, Hanie Sedghi, Olga Saukh, and Behnam Neyshabur. 2021. The role of permutation invariance in linear mode connectivity of neural networks. *arXiv preprint arXiv:2110.06296* (2021).

[13] Ahmad Esmaeili, Zahra Ghorrati, and Eric T Matson. 2023. Holonic learning: A flexible agent-based distributed machine learning framework. *arXiv preprint arXiv:2401.10839* (2023).

[14] Liang Feng, Abhishek Gupta, Kay Chen Tan, and Yew Soon Ong. 2023. *Evolutionary Multi-Task Optimization.* Springer.

[15] Timur Garipov, Pavel Izmailov, Dmitrii Podoprikhin, Dmitry P Vetrov, and Andrew G Wilson. 2018. Loss surfaces, mode connectivity, and fast ensembling of dnns. *Advances in neural information processing systems* 31 (2018).

[16] Harsh Goel, Yifeng Zhang, Mehul Damani, and Guillaume Sartoretti. 2023. Sociallight: Distributed cooperation learning towards network-wide traffic signal control. *arXiv preprint arXiv:2305.16145* (2023).

[17] Joseph Henrich and Richard McElreath. 2003. The evolution of cultural evolution. *Evolutionary Anthropology: Issues, News, and Reviews: Issues, News, and Reviews* 12, 3 (2003), 123–135.

[18] Charles Isbell, Christian R Shelton, Michael Kearns, Satinder Singh, and Peter Stone. 2001. A social reinforcement learning agent. In *Proc. 5th Int. Conf. on Autonomous Agents.* 377–384.

[19] Pavel Izmailov, Dmitrii Podoprikhin, Timur Garipov, Dmitry Vetrov, and Andrew Gordon Wilson. 2018. Averaging weights leads to wider optima and better generalization. *arXiv preprint arXiv:1803.05407* (2018).

[20] Mattia Jacopo Villani, Emanuele Natale, and Frederik Mallmann-Trenn. 2025. Trading-off Accuracy and Communication Cost in Federated Learning. *arXiv e-prints* (2025), arXiv–2503.

[21] Wenzheng Jiang, Ji Wang, Xiongtao Zhang, Weidong Bao, Cheston Tan, and Flint Xiaofeng Fan. 2025. FedHPD: Heterogeneous Federated Reinforcement Learning via Policy Distillation. *arXiv preprint arXiv:2502.00870* (2025).

[22] Timo Kaufmann, Paul Weng, Viktor Bengs, and Eyke Hüllermeier. 2024. A survey of reinforcement learning from human feedback. (2024).

[23] Rachel L Kendal, Neeltje J Boogert, Luke Rendell, Kevin N Laland, Mike Webster, and Patricia L Jones. 2018. Social learning strategies: Bridge-building between fields. *Trends in cognitive sciences* 22, 7 (2018), 651–665.

[24] W Bradley Knox and Peter Stone. 2011. Augmenting reinforcement learning with human feedback. In *ICML 2011 workshop on new developments in imitation learning (July 2011)*, Vol. 855.

[25] Dmytro Kuzmenko and Nadiya Shvai. 2025. Knowledge Transfer in Model-Based Reinforcement Learning Agents for Efficient Multi-Task Learning. *arXiv preprint arXiv:2501.05329* (2025).

[26] Yuyu Li and Jianmin Ji. 2021. Parallel curriculum experience replay in distributed reinforcement learning. In *Proceedings of the 20th International Conference on Autonomous Agents and MultiAgent Systems.* 782–789.

[27] Adi Livnat. 2013. Interaction-based evolution: how natural selection and nonrandom mutation work together. *Biology direct* 8 (2013), 1–53.

[28] Adi Livnat and Christos Papadimitriou. 2016. Sex as an algorithm: the theory of evolution under the lens of computation. *Commun. ACM* 59, 11 (2016), 84–93.

[29] Donald Loveland, James Usevitch, Zachary Serlin, Danai Koutra, and Rajmonda Caceres. 2025. MAGNET: A Multi-Agent Graph Neural Network for Efficient Bipartite Task Assignment. In *Proceedings of the 24th International Conference on Autonomous Agents and Multiagent Systems.* 1399–1407.

[30] Kamal K Ndousse, Douglas Eck, Sergey Levine, and Natasha Jaques. 2021. Emergent social learning via multi-agent reinforcement learning. In *ICML.* 7991–8004.

[31] Robert G Reynolds (Ed.). 2020. *Cultural algorithms: tools to model complex dynamic social systems.* John Wiley & Sons.

[32] Sebastian Risi, Yujin Tang, David Ha, and Risto Miikkulainen. 2025. *Neuroevolution: Harnessing Creativity in AI Model Design.* MIT Press. https://neuroevolutionbook.com/.

[33] Alejandro Romero, Gianluca Baldassarre, Richard J Duro, and Vieri Giuliano Santucci. 2023. Learning multiple tasks with non-stationary interdependencies in autonomous robots. In *Proceedings of the 2023 International Conference on Autonomous Agents and Multiagent Systems.* 2547–2549.

[34] Ghada Sokar, Decebal Constantin Mocanu, and Mykola Pechenizkiy. 2021. Self-attention meta-learner for continual learning. *arXiv preprint arXiv:2101.12136* (2021).

[35] George Stoica, Daniel Bolya, Jakob Bjorner, Pratik Ramesh, Taylor Hearn, and Judy Hoffman. 2023. Zipit! merging models from different tasks without training. *arXiv preprint arXiv:2305.03053* (2023).

[36] Derek Tam, Margaret Li, Prateek Yadav, Rickard Brüel Gabrielsson, Jiacheng Zhu, Kristjan Greenewald, Mikhail Yurochkin, Mohit Bansal, Colin Raffel, and Leshem Choshen. 2024. Llm merging: Building llms efficiently through merging. In *NeurIPS 2024 Competition Track.*

[37] Peng Tang, Lifan Wang, Weidong Qiu, Zheng Huang, and Qiangmin Wang. 2024. Fuzzy clustered federated learning under mixed data distributions. In *Proceedings of the 23rd International Conference on Autonomous Agents and Multiagent Systems.* 2501–2503.

[38] Daniel J van der Post, Mathias Franz, and Kevin N Laland. 2016. Skill learning and the evolution of social learning mechanisms. *BMC evolutionary biology* 16 (2016), 1–19.

[39] Gregory M Werner. 1991. Evolution of communication in artificial organisms. *Artificial Life* (1991).

[40] Mitchell Wortsman, Gabriel Ilharco, Samir Ya Gadre, Rebecca Roelofs, Raphael Gontijo-Lopes, Ari S Morcos, Hongseok Namkoong, Ali Farhadi, Yair Carmon, Simon Kornblith, et al. 2022. Model soups: averaging weights of multiple fine-tuned models improves accuracy without increasing inference time. In *International conference on machine learning.* PMLR, 23965–23998.

[41] Haotong Zhang and Wanyuan Wang. 2025. Fast Adaption by Policy Deviation Integral Meta-reinforcement Learning with Applications to High-speed Trains Operation. In *Proceedings of the 24th International Conference on Autonomous Agents and Multiagent Systems.* 2819–2821.

[42] Luisa Zintgraf, Sam Devlin, Kamil Ciosek, Shimon Whiteson, and Katja Hofmann. 2021. Deep interactive bayesian reinforcement learning via meta-learning. *arXiv preprint arXiv:2101.03864* (2021).