COUPLING SEMI-SUPERVISED LEARNING WITH REINFORCEMENT LEARNING FOR BETTER DECISION MAKING — AN APPLICATION TO CRYO-EM DATA COLLECTION

Anonymous authors
Paper under double-blind review

ABSTRACT

We consider a semi-supervised Reinforcement Learning (RL) approach that takes inputs of a perception model. Performances of such approaches can be significantly limited by the quality of the perception model in the low labeled data regime. This paper proposes a novel iterative framework that simultaneously couples and improves the training of RL and the perception model. The perception model takes pseudo labels generated from the trajectories of a trained RL agent believing that the decision-model can correct errors made by the perception model. We applied the framework to cryo-electron microscopy (cryo-EM) data collection, whose goal is to find as many high-quality micrographs taken by cryo-electron microscopy as possible by navigating at different magnification levels. Our proposed method significantly outperforms various baseline methods in terms of both RL rewards and the accuracy of the perception model. We further provide some theoretical insights into the benefits of coupling the decision model and the perception model by showing that RL generated pseudo labels are biased towards localization which aligns with the underlying data generating mechanism. Our iterative framework that couples both sides of the semi-supervised RL can be applied to a wide range of sequential decision-making tasks when the labeled data is limited.

1 INTRODUCTION

Decoupling representation learning or perception learning from Reinforcement Learning (RL) is commonly used to improve performance in RL applications (Stooke et al., 2021). For example, the idea of state abstraction for RL concerns learning a low dimensional state representation to deal with a large state space (Jong & Stone, 2005; Abel et al., 2016; Raffin et al., 2018; Ho, 2019). The success of decoupling perception learning from RL depends on the quality of the perception model, which often requires a large amount of labeled data for training. In many realistic scenarios, acquiring fully labeled datasets is nevertheless costly and sometimes infeasible, while acquisition of unlabeled data is relatively inexpensive. Such situations render semi-supervised learning (SSL) (Zhu, 2005) a natural choice for obtaining good perception representations with limited annotations for RL. However, a naive application of SSL to perception models may not necessarily lead to promising results for RL because a) the improvement of SSL in the case of a small number of labeled data can be too subtle to facilitate RL; and b) the improved overall accuracy of the perception model may not be directly relevant to better RL policies.

Interestingly, in many cases, an RL agent can provide useful feedback to the perception model through the quality of sampled trajectories during learning. We investigate the idea of improving perception modeling by RL under an SSL setting with limited labeled data and vice versa. We specifically consider a family of navigation problems with the goal of discovering as many targets of interest as possible. For example, Scavenger hunt (Yedidsion et al., 2021) trains a robot to search places with high probability of finding the targets, and Fan et al. (2022) applies RL to optimize microscope movement for efficient CryoEM data collection. The structure of such navigation problem permits a straightforward approach to generates pseudo labels directly from current policies' rollouts and correct mistakes made by the perception model, as illustrated by the example in Figure 1.

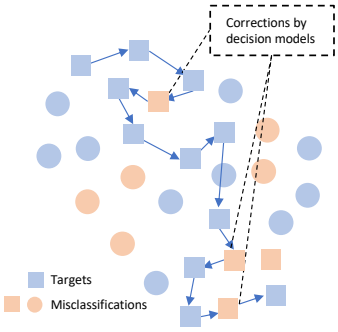


Figure 1: A simple path planning toy example of RL trajectories correcting mistakes made by the perception model. The goal of the agent to find as many targets of interest as possible (marked by squares). A pretrained classifier is used for the perception model, whose misclassificaitons are marked by orange The misclassifications of the model are marked by orange.

In light of the intuition above, we propose to couple perception modeling and RL in an iterative framework to mutually enhance each other in scenarios with label shortage issues. Specifically, we leverage state trajectories sampled from a learned RL policy to generate pseudo labels to improve the perception model. The improved perception representations, which, in turn, provide better input to RL, lead to more effective RL polices. We alternate perception modeling and RL iteratively until converge. Since both perception modeling and RL use labeled and unlabeled data for training, we dub our approach *SSL*^{2}*-RL* (SSL-RL with SSL learned perception models).

SSL has been applied to improve RL (SSL-RL) where the reward function can only be evaluated in some settings but not all. For instance, Finn et al. (2016) uses unlabeled trajectories for a better importance sampling estimator of a particular parameter in the entropy objective function. Konyushkova et al. (2020) learns a reward function to annotate the trajectories without generating new unlabeled trajectories to improve the reward function. Fu et al. (2017) learns a discrimination model to discriminate the RL trajectories from the positive examples in a binary reward setup. The major difference between our approach and Sem fin the literature is that our approach directly generates pseudo labels from RL, while other approaches either utilizes the unlabeled trajectories in an indirect way or uses a fixed reward model without trying to leverage the feedbacks from RL for a better reward modeling.

1.1 Related literature

Label propagation. Label propagation propagates labels through a dataset along high density areas defined by unlabeled data. It follows the intuition that close points should have similar labels. Zhu & Ghahramani (2002) iteratively propagates labels using a linear combination of adjacent nodes defined on a graph. Su et al. (2015); Vernaza & Chandraker (2017); Jabri et al. (2020) perform label propagation through random walk. Our method can be seen as a special way of propagating labels through a decision-making models, which incorporates both context information and the geometric information. Cai et al. (2021) proposed to optimize the loss with a regularization on the inconsistency over samples within the same neighborhood.

Semi-supervised RL. Semi-supervised RL concerns the problem where the agent must perform RL when the reward function is known in some settings, but cannot be evaluated in others. For semi-supervised RL, a wide range of pseudo reward is generated. For example, Finn et al. (2016); Fu et al. (2018); Singh et al. (2019); Konyushkova et al. (2020) learns a classifier for reward labeling using a labeled dataset, which is applied to optimize a entropy-regularized objective for an unlabeled dataset. Yu et al. (2022) states that a zero pseudo reward is sufficient for tasks using sparse reward functions. Some other pseudo rewards are proposed using task-specific prior knowledge such as distance to goals in goal-conditioned settings (Andrychowicz et al., 2017). In contrast to others, some approaches directly imitate expert trajectories to achieve high-levels of performance without requiring reward labels (Ross & Bagnell, 2012; Ho & Ermon, 2016).

Cryo-EM data collection. We have focused this work on addressing the issue of cryo-EM data collection. Cryo-EM serves as a critical tool for determining the three-dimensional structures of biological macromolecules. As such, cryo-EM is a powerful tool in the development of vaccines and therapeutics to combat diseases such as COVID-19. Within weeks of the release of the genomic sequence of SARS-CoV-2, cryo-EM determined the first SARS-CoV-2 spike protein structure (Wrapp et al., 2020). Since this original publication, cryo-EM was used to determine additional SARS-CoV-

2 structures such as spike protein bound to antibody fragments (Lempp et al., 2021; Scheid et al., 2021), remdesivir bound to SARS-CoV-2 RNA-dependent RNA polymerase (Bravo et al., 2021; Yin et al., 2020; Kokic et al., 2021), and reconstructions of intact SARS-CoV-2 virions (Yao et al., 2020; Ke et al., 2020).

2 PROBLEM FORMULATION

We study the problem of training RL policies for navigation, where there are a large amount of unlabeled data whereas very few labels are available for learning perception representations for RL. We start by defining the RL environment, which consists of four major elements, i.e the state space, action space, transition function and reward function. The state space is a set of tuples, where each state is denoted by (s, x, y), where $s \in \mathcal{S}$ encodes the context information that uniquely identifies each state, $y \in \{0, 1\}$ is a binary label and, $x \in \mathcal{X}$ is the input feature that can be used to predict y. Depending on the actual application, the context can be interpreted as the geometric information that encodes the location of the state on a map. Whenever is clear from the context, we let $y(s)$ be the true label corresponding to the state s. At the step t the agent is provided with an action set $\mathcal{A}_t$ and the next reward and state are sampled from the $R : \mathcal{S} \times \mathcal{A} \mapsto [0, 1]$ and $T : \mathcal{S} \times \mathcal{A} \mapsto \mathcal{S}$. We consider a deterministic transition function. Throughout the paper, we consider a reward function that is directly relevant to the labels, i.e., $R(s, a) = \mathbb{1}(y(s) = 1) + c(s, T(s, a))$, where $c : \mathcal{S} \times \mathcal{S} \mapsto \mathbb{R}$ is a cost function. In the context of navigation problem, c prevents the agent from conducting large movements.

We consider a semi-supervised learning scenario for both the perception model and Reinforcement Learning. We are given both labeled and unlabeled dataset denoted by $\mathcal{L} = \{s_i, x_i, y_i\}_{i=1}^{N_L}$ and $\mathcal{U} = \{s_i, x_i\}_{i=1}^{N_U}$, respectively, where N_L and N_U are their sizes. The perception model is a mapping $f : \mathcal{X} \mapsto [0, 1]$ that predicts the positive label probability with input feature x. Note that we consider a binary label for easier presentation, while our framework can be extended to the multi-class case.

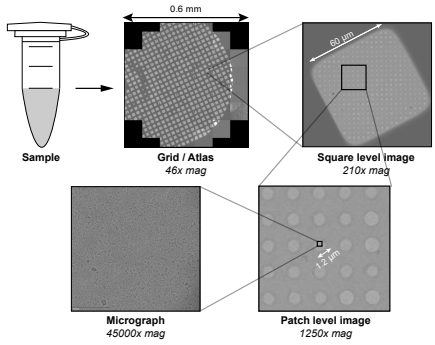
2.1 CRYO-EM DATA COLLECTION

Cryo-EM is a key technique for structural biology that enables 3D structure determination of important macromolecular complexes and membrane proteins Wrapp et al. (2020). Cryo-EM data collection involves steering transmission electron microscopes hierarchically at different magnification levels (as shown in Figure 2) to explore a grid with the goal of identifying and collecting high-quality micrographs at high magnification. This sequential process includes several mechanical operations to allow microscope navigation to different regions of a grid, namely grid switching, square switching, and patch switching. An effective data collection session aims at finding a sequence of holes where there is a considerable portion of high-quality micrographs. However, it is a highly involved and time-consuming process that requires expertise and skills to make decisions at different levels of microscope operations.

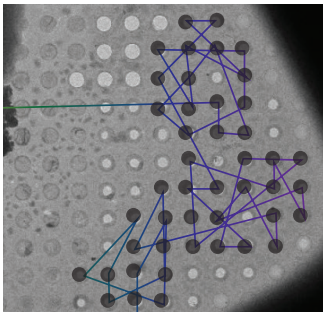
To mitigate that inefficiency in data collection, Fan et al. (2022) proposed to train an RL agent for a automatic cryo-EM data collection. Their framework is called cryoRL, which first trains an image classifier. The predictions of the image classifier as well as the distributions of labels within each grid, square, patch are used as features to train a DQN agent. They maximize the total number of high quality holes within a fixed budget of time, which cast requirement on a efficient path that does not need to steer the microscopes too frequently. Similar practice can also be found in Li et al. (2022).

CTF (contrast transfer function) is used to evaluate the quality of a hole. As a variable evaluated on the multiple micrographs for each hole, it is infeasible to evaluate CTFs for all the holes in a dataset before navigation. In real data collection practice, a numerous number of samples are generated in the daily data collection practice, while only a small proportion of them can be actually evaluated and labeled, which makes cryo-EM data collection a perfect example to apply SSL approaches.

To fit cryo-EM data collection into the proposed problem formulation, we let each hole in the cryo-EM data collection be a state in the problem state. A hole can be represented as $\{s_i, x_i, y_i\}$, where $s_i = (\text{grid}_i, \text{square}_i, \text{patch}_i)$ represents grid, square and patch indices of the i-th hole. x_i is the hole-level image of the i-th hole and y_i represents the true quality of the hole.



(a)



(b)

Figure 2: (a) (Figure 2 in Fan et al. (2022)) Overview of cryo-EM data collection. A purified sample is prepared and vitrified on the support grid. The atlas image provides a low magnification overview by stitching multiple "grid-level" images into a single montage. Next, users will select specific squares to image at medium magnification. After inspection, the user selects "patch" areas on the square to inspect holes with higher magnification, using the patch image to decide holes to collect for micrographs. The micrographs contain high-resolution images for downstream data processing. (b) (Adapted from Li et al. (2022)) A trajectory collected by a trained RL policy.

3 PROPOSED METHOD

We first discuss the training of the perception model and the RL model separately, before the iterative algorithm that integrates the two models are presented.

Perception model training. The quality of the perception model determines the overall quality of the RL agent. We propose to use the semi-supervised learning method, i.e. *FixMatch* (Sohn et al., 2020). *FixMatch* adds an unsupervised loss that regularizes the inconsistency between the strongly augmented and weakly augmented inputs from the unlabeled dataset. Recall that $f : \mathcal{X} \mapsto [0, 1]$ is the perception model. We let $P_f : \mathcal{X} \mapsto [0, 1]^2$ be the predicted label distribution over $\{0, 1\}$. The unsupervised loss is given by

$$l_U(f) = \sum_{i=1}^{N_U} \mathbb{1}(\max\{P_f(x_i^w)\} \geq \tau) H\left(P_f(x_i^w), P_f(x_i^s)\right), \tag{1}$$

where x_i^w, x_i^s are the weakly and strongly augmented inputs of the i-th input in unlabeled data, and H is the entropy function between two distributions.

In the cryo-EM task, we solve an binary image-classification problem, using a CTF threshold 6. As seen in Table 1b), the performance of a supervised model trained from the fully labeled dataset is $\sim 65\%$ only, indicating the classification task is nontrivial. A shown in Fig. 7 of the Appendix B, with a cutoff threshold 6.0, many samples in the hole data lie around the threshold, suggesting that the training data is quite ambiguous.

RL policy training. Since the labeled data can be highly limited and training of RL is unstable with a small number of observations, we train RL on both labeled and unlabeled data to utilize the information from the unlabeled data. As the pretrained classifier can be seen as a prediction on reward function (without movement cost), it is natural to follow the commonly used approach that generates pseudo rewards through predicted reward labels (Finn et al., 2016). Let the reward at the step t be $\tilde{r}_t = y_{t+1}\mathbb{1}(s_{t+1} \in \mathcal{L}) + f(x_{t+1})\mathbf{1}(s_{t+1} \in \mathcal{U}) - c(s_t, s_{t+1})$. For instance, in cryo-EM task, whenever a hole in $\mathcal{L}$ is visited, a reward is generated from the true labels. If the hole is unlabeled, a pseudo reward is given by the predicted probability of being low CTF. Following Fan et al. (2022), we add to the final rewards an extra cost function that penalizes large movements (See Appendix A for details). We consider a constrained RL, which terminates an episode whenever the cumulative cost $c(s_t, s_{t+1})$ reaches a threshold τ. With the pseudo rewards, we train a regular offline DQN on

the whole dataset (Van Hasselt et al., 2016) to optimize the following objective function:

$$\max \sum_{l=0}^{n_l} \tilde{r}_l \quad \text{s.t.} \sum_{l=0}^{n_l} c(s_t, s_{t+1}) \leq \tau, \text{ where } n_l \text{ is the index when terminated.}$$

Pseudo labels for perception models. Since we consider a navigation problem, where the visits of trajectories directly indicate the chances of find a target of interest, we generate pseudo labels straightly from the visiting orders of trajectories as opposed to other SSL-RL methods (Finn et al., 2016; Fu et al., 2017). This approach allows us to back-propagate the geometric structural bias learned by RL agent back to the perception model. Let $(S_1, \ldots, S_{N_U+N_L})$ be the sequence of states the policy iterates until all the states are visited. For a given cutoff $N_C > 0$, we label the first N_C states as positive while the rest of states negative. Note although we evaluate on the whole dataset, we will only use the pseudo labels for unlabeled data. A visualization of the process can be found in the right panel of Figure 3. As the starting point are chosen in a stochastic way, we evaluate the policy for M independent times, which gives M pseudo labels for each state for a more robust labeling process. Let the pseudo labels for the i-th state in the m-th run be $\bar{Y}_{im}$. Let the pseudo label $\bar{Y}_i$ be the majority of $\{\bar{Y}_{i1}, \ldots, \bar{Y}_{im}\}$. We let the confidence of each pseudo label be $p_i = \sum_{m=1}^{M} \bar{Y}_{im}/M$ if $\bar{Y}_i = 1$ and $1 - \sum_{m=1}^{M} \bar{Y}_{im}/M$ if $\bar{Y}_i = 0$.

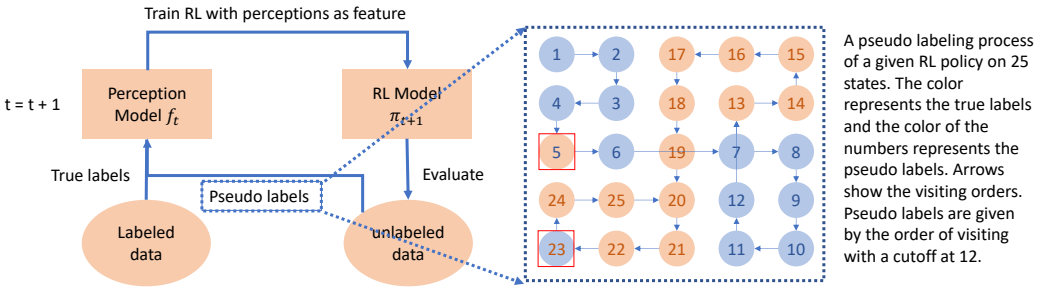


Figure 3: An iterative semi-supervised framework for perception and RL models. On the round t, the framework trains a RL agent π_{t+1} that takes the perception model f_t as input. By evaluating the agent $t+1$ on the unlabeled dataset, it generates the pseudo labels for each visited state. The perception model at the next step is trained on both labeled dataset with true labels and unlabeled dataset with pseudo labels.

Iterative framework. Our main idea to integrate RL and perception learning. We propose to feed the pseudo labels back to the perception model. We fine-tune the pretrained classifier on the whole dataset using pseudo labels for the unlabeled data. Each input in the unlabeled data is sampled with a probability proportional to its confidence. Let $\text{CE} : [0,1]^2 \times \{0,1\} \mapsto \mathbb{R}$ be the cross entropy loss. The loss function of fine-tuning the perception model with soft pseudo labels from RL trajectories is then given by $l(f) = l_{\text{sup}}(f) + \lambda l_{\mathcal{U}}(f)$, where

$$l_{\text{sup}} = \sum_{i=1}^{N_U+N_L} p_i \, \text{CE}(f(x_i), \bar{Y}_i) \mathbb{1}(s_i \in \mathcal{U}) + \text{CE}(f(x_i), y_i) \mathbb{1}(s_i \in \mathcal{L}).$$

An overview of our propose method is given in Algorithm 1. It may not reach the best performance in one-round. Thus, we repeat the above process for multiple rounds. The round with the best validation performance is selected as the final model.

4 Experimental Results

In this section, we first introduce some implementation details, and then present the experimental results on Cryo-EM dataset. Most of our implementations follow the setups in Fan et al. (2022). We briefly go through important details, while referring the readers to the Appendix A for the complete setups.

Algorithm 1 Iterative framework for the joint training of the perception and RL models

Input: Labeled and unlabeled dataset $\mathcal{L}, \mathcal{U}$ and the number of iterations $\mathcal{K}$
Pretrain teacher classifier f_0 on $\mathcal{L}$ and $\mathcal{U}$ using *FixMatch* .
for $t = 1, \dots \mathcal{K}$ **do**
  Train RL on both $\mathcal{L}$ and $\mathcal{U}$ with pseudo rewards predicted by classifier f_{t-1}.
  Generate pseudo labels $\hat{Y}_1, \dots, \hat{Y}_{N_L+N_U}$.
  Fine-tune the classifier f_{t-1} with the pseudo labels which generates f_t.
end for

We experiment on a cryo-EM dataset called Y3 with 8653 holes over 9 grids, 58 squares and 771 patches. We split the dataset into training and validation dataset with 6489 and 2164 hole respectively. Each hole corresponds to a state in the environment. The feature information for each state is the hole-level image observation. Note that the ground truth CTFs are valued by the micrographs, which can not be accessed through hole-level images.

We train a ResNet-18 (He et al., 2016) to classify the hole-level images for the perception model. Hyperparameters for *FixMatch* training is given in Appendix A.

Apart from the hole-level predictions from the perception model, we add the following features to encode the geometric information for RL policy training. For each of the patch, square and grid, we compute the number of unvisited holes, unvisited low CTFs holes, visited holes and visited low CTFs hole within the patch, square and grid, respectively. Additionally, we have three dummy variables encoding whether the agent reaches a new patch, square or grid. During the training, the features of the past three steps are concatenated as the input of DQN. We terminate an episode whenever the duration, i.e. the cumulative sum of cost reaches certain threshold. Two thresholds 120 and 480 are considered for RL training. We use a three layer MLP model with hidden sizes (128, 256, 128) and ReLU activation function for the Q-network.

4.1 RESULTS

We experiment on 5%, 10% and 20% of the training data and conduct evaluation on the entire validation set. We compare our proposed approach with 3 baseline methods: a) the cryoRL method proposed in Fan et al. (2022) based on a supervised classifier (***SL***); b) cryoRL based on *FixMatch* (***FixMatch***); and c) cryoRL based on ***Iterative FixMatch*** that runs FixMatch multiple rounds with pseudo labeling (provided by the perception model obtained in the last round used for cryoRL). We evaluate our proposed method for duration 120 and 480 (i.e. ***SSL*^{2}*-RL* 120** and ***SSL*^{2}*-RL* 480**), respectively. For fairness, cryoRL is trained with both labeled and unlabeled data in all cases and evaluated at a duration of 480. The RL rewards and the accuracy of the corresponding perception models are presented in Table 1. For algorithms that do iteration, the best validation RL rewards and the corresponding accuracy are presented. For reference, when the fully labeled dataset is used, the classification model achieves an accuracy of 65.24%, and the best RL reward from cryoRL is 69.76.

With 5% of the labeled data, *FixMatch* improves the classification accuracy by 5% compared with supervised learning. By further increasing the labeled data to 10% and 20%, the improvement ($\sim$ 1%) becomes less obvious. Our proposed approach ($SSL^2 - RL480$) consistently outperforms FixMatch by $\sim$ 2% and is on par with the supervised model trained using 100% labeled data (65.24%). As a comparison, iterative FixMatch performs only slightly better than FixMatch, clearly indicating the effectiveness of incorporating feedback from RL. By incorporating feedbacks from RL (*SSL*^{2}-*RL* 120. A similar trend can also be found in terms of the RL rewards (Table 1b), suggesting that RL benefits from improved classification overall. Iterative approaches performs in general better than the non-iterative approaches. Nevertheless, *SSL*^{2}*-RL* 480 still outperforms *FixMatch* +iteration by 2. Figure 4 (b) visualizes the total number of low-CTF holes found by different approaches. Not surprisingly, our approach outperforms all the others. As shown in Figure 8, the quality of pseudo labels is able to generate labels at a higher accuracy then the that of the perception model at the current round.

Classification accuracy is not the best metric to reflect the performance. We further investigate the precision score. Precision score measures the number of true positive out of all the positive samples predicted by the model. It is more consistent with the RL rewards since an episode is

Table 1: A summary of RL rewards and classification accuracy of compared methods. Table (a) shows the classification accuracy for the perception model. For the iterative methods, we report the results that reaches the highest RL reward over 10 independent runs. Table (b) shows the average RL rewards and their standard deviation. Bold text marks the best RL rewards for each row.

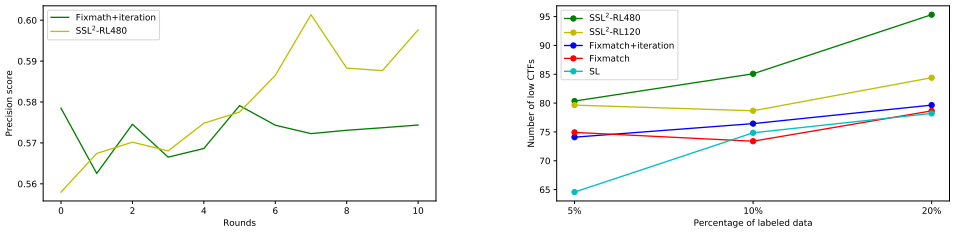
(a) Classification accuracy

% of labels	*SL*	*FixMatch*	*Iterative FixMatch*	*SSL*^{2}*-RL* 120	*SSL*^{2}*-RL* 480
5%	0.5707	0.6229	0.6372	0.6423	**0.6451**
10%	0.6188	0.6303	0.6377	0.6480	**0.6557**
20%	0.6299	0.6382	0.6396	**0.6502**	0.6479
100%	0.6524	-	-	-	-

(b) RL rewards

% of labels	*SL*	*FixMatch*	*Iterative FixMatch*	*SSL*^{2}*-RL* 120	*SSL*^{2}*-RL* 480
5%	59.55 ± 5.4	56.97 ± 3.2	62.33 ± 7.5	**62.94** ± 4.6	61.62 ± 7.1
10%	50.96 ± 5.6	58.50 ± 5.5	61.95 ± 3.4	64.28 ± 8.5	**65.73** ± 7.0
20%	56.76 ± 7.3	58.98 ± 3.5	65.77 ± 4.2	64.29 ± 8.2	**67.28** ± 6.3
100%	69.76 ± 2.1	-	-	-	-

terminated at duration 480, which only allows the agent to visit a small number of holes. In Figure 4 (a), we observe significant increases in the precision score during the iteration, which marks the overall improvement in the quality of the perception models, while the compared iteration method, *FixMatch* +iteration does not show a similar improvement in precision score during the iteration. Another metric is to directly compare the number of low-CTF holes found by the trained RL agent. As shown in Figure 4 (b) SSL-RL has the dominant performance over other methods at different levels of percentages of labeled data.



(a) Precision score

(b) Number of low-CTF holes

Figure 4: (a) Changes in precision score over 10 rounds of iteration for *SSL*^{2}*-RL* 480 and *FixMatch* +iteration for 10% of labaled data. (b) The average number of low CTF holes found by the trained RL agents within 480 duration for different methods under different percentages of labeled data.

4.2 Ablation Study

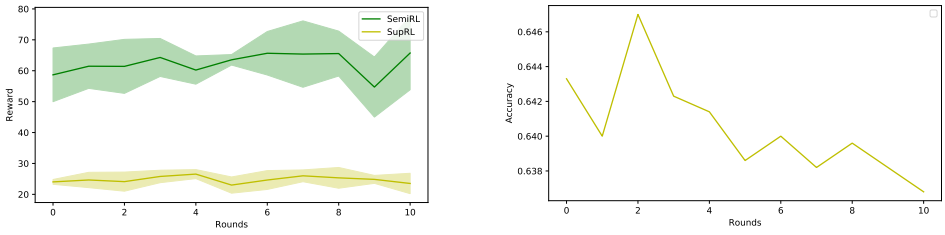
In this section, we conduct experiments to characterize the proposed approach. We investigate the following components including an alternative way to select models for iteration methods, importance of using semi-supervised learning for RL, the essential of using cost penalty inside the iteration and the performance of other RL methods.

Termination strategy. In table 1, we compare the results of iteration approaches that terminate when RL reward is the highest. One can also terminate when classification accuracy reaches the highest. The results are given in Appendix B Table 3, which is similar to Table 1.

Without Semi-supervised RL. We remove the use of pseudo rewards for RL and use only 10% of the data to train RL policies. Figure 5 (a) shows the change of RL rewards during a 10-round

iteration. There is a significant gap between semi-supervised RL and supervised RL that trains only on the 10% labeled data.

Without moving cost penalty. Though we will show later that the movement cost introduces strong bias towards localization, which may improve the quality of pseudo labels, we empirically investigate the benefits of adding movement penalty. We can see that the classification accuracy does not increase. Figure 5 (b) shows the change of classification accuracy of *SSL*^{2}*-RL* 480 for a 10-round iteration on 10% data. We don't see significant increase on classification accuracy. It also performs worse than the results reported in Table 1.



(a) Semi-supervised RL vs. supervised RL

(b) Acc. for *SSL*^{2}*-RL* 480 without movement cost

Figure 5: (a) RL rewards for 10-rounds *SSL*^{2}*-RL* 480 with policies trained by semi-supervised RL and supervised RL, respectively. (b) Classification accuracy for 10-round *SSL*^{2}*-RL* 480 without movement penalty on 10% of labeled data.

Other RL models. DQN is used for decision-models in Table 1. We replace DQN with other RL models, e.g. A2C Rosenstein et al. (2004) and Rainbow. The best RL rewards and classification accuracy by *SSL*^{2}*-RL* 480 using A2C over 10-round iterations are 61.80 ± 5.7 and 0.64 respectively for 10% of data. These of Rainbow are 62.80 ± 3.1 and 0.64. Both A2C and Rainbow are worse than *SSL*^{2}*-RL* 480 using DQN. Note that this is also consistent with the observations in Fan et al. (2022).

Table 2: Performances of Rainbow and A2C compared with DQN

Metrics	SSL^2-RL 120	SSL^2-RL 480	Rainbow 480	A2C 480
Accuracy	0.6557	0.6480	0.6400	0.6430
RL rewards	64.28+9.5	65.73+7.0	62.80+3.1	61.80+5.7

5 THEORETICAL UNDERSTANDING

In this section, we provide some theoretical insights into the benefits of our proposed method. A key to understanding our problem is whether RL could generate better pseudo labels than the classifier pretrained on the labeled dataset. Recall the pseudo label of the i-th state is denoted by $\bar{Y}_i$. We study whether $\sum_{i=1}^{N_U} \mathbb{1}(\bar{Y}_i = y_i) \geq \sum_{i=1}^{N_U} \mathbb{1}(f(x_i) = y_i)$.

Benefits of RL label propagation. Pseudo labels from RL can be seen as a special way of doing label propagation. As opposed to label propagation through random walk, RL navigate on the map under the guide of the pretrained predictor. We understand the benefits of using RL for label propagation in the following two ways.

First, RL agents are trained with additional geometric information. It is normally not easy for a classifier to encode geometric information since most classifiers treats data i.i.d. For example, if the input features are images, popular image classification models does not directly incorporate dependence among images. Second, the movement costs added to the RL reward function induce bias towards localization of the true labels, which may align with the true label generating process. To this end, we use the example in Figure 6 to illustrate. The RL trajectory is able to correctly classify cluster 4 (on the right panel), because starting from 3 RL tries to avoid large movements.

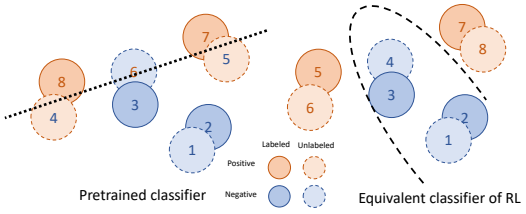


Figure 6: An illustration of localization bias from RL pseudo labels (adapted from Cai et al. (2021)). The black lines represent the (equivalent) decision boundaries of the pretrained classification model and RL. The numbers represents the visiting orders. The equivalent decision function by RL achieves 100% accuracy due to the localization.

Localization of RL-based Label propagation. In this section, we rigorously discuss the localization property of the train RL policy. To this end, we introduce some extra setups. We assume that the marginal distribution of (s, x) is L for the labeled dataset and U for the unlabeled dataset. Let the inconsistency rate between two predictors $g_1, g_2 : \mathcal{S} \times \mathcal{X} \mapsto \mathcal{Y}$ be $\mathcal{E}^L(g_1, g_2) = \mathbb{E}_{s,x\sim L}\mathbb{1}(g_1(s, x) \neq g_1(s, x))$.

In the literature, label propagation is given by regularizing the consistency across neighboring points. Cai et al. (2021) proposes to solve the following optimization problem for a improved classifier f^*:

$$f^* = \underset{f:\mathcal{S}\times\mathcal{X}\to\mathcal{Y}}{\operatorname{argmin}} \mathcal{E}^L(f, f_{tc}) \text{ s.t. } R_{\mathcal{B}}(g) \leq \mu, \text{ for some } \mu > 0, \tag{2}$$

where f_{tc} is the pretrained classifier and the regularization is defined by

$$R_{\mathcal{B}}(f) = \mathbb{P}_{s,x\sim\frac{1}{2}(L+U)}\left[\exists s' \in \mathcal{B}(s) \text{ , s.t. } f(s', x) \neq f(s, x)\right],$$

and $\mathcal{B}(s)$ is the neighboring of s. In practice, one optimizes its empirical version.

Now we define the object of RL training. For a trained policy π, let $(S_1, Y_1, \ldots, S_{N_U+N_L}, Y_{N_U+N_L})$ be the trajectory of visited state and labels by evaluating π on the whole dataset. We aim at finding the policy π that maximizes the regularized cumulative rewards up to step N_C:

$$\pi^* = \operatorname{argmax}_\pi \sum_{t=1}^{N_U+N_L} \mathbb{1}(Y_t = 1)\mathbb{1}(t \leq N_C) - c(S_t, S_{t+1})^1. \tag{3}$$

Slightly abusing the notation, we let the equivalent decision boundary of a policy π by $f_\pi : \mathcal{S}\times\mathcal{X} \mapsto \{0, 1\}$, such that $f_\pi(s, x) = \mathbb{1}(t(s, \pi) \leq N_C)$, where $t(s, \pi)$ is the step in which s being visited by running policy π. We have the following lemma that proves the equivalence between the two regularization.

Lemma 1. *Let $\{\mathcal{S}_1 \ldots, \mathcal{S}_B\}$ be a B-partition of $\mathcal{S}$, i.e. $\cup_{b=1}^B \mathcal{S}_b = \mathcal{S}$ and let $P(s)$ be the partition s belongs to. We define the neighbor function by the partitions, i.e. $\mathcal{B}(s) = \{s' \in \mathcal{S} : P(s') = P(s)\}$. Then for all policy π, $\hat{\mathcal{R}}_{\mathcal{B}}(f_\pi) \leq C_1 \sum_t c(S_t, S_{t+1}) + C_2$, for some universal constants C_1 and C_2. The equality holds, if π visits each partition at most twice.*

Cai et al. (2021) shows that the improved classifier can achieve arbitrary small classification error even if the error rate of the pretrained classifier is high.

6 Discussion and Limitations

In this paper, we proposed *SSL^2-RL* , an iterative framework that joint learns the perception model and decision-making model. We focus on the navigation problem, which allows us to connect the learning of RL and that of the perception model by directly generating pseudo labels from trajectories. The framework shows significant improvements in cryo-EM data collection task. We then showed that RL with a penalty on large movement induces bias towards localization on the pseudo labels, which may improve the quality of the pseudo labels. A potential direction is to extend the framework to more general RL problems. Currently our approach only applies to navigation problem where the orders of visiting in a trajectory imply the labels for the perception learning. One potential way to generalize is to generate pseudo labels from the learned Q function.

[1]Note that we consider the policy running through the whole dataset even after it is terminated

REFERENCES

David Abel, David Hershkowitz, and Michael Littman. Near optimal behavior via approximate state abstraction. In *International Conference on Machine Learning*, pp. 2915–2923. PMLR, 2016.

Marcin Andrychowicz, Filip Wolski, Alex Ray, Jonas Schneider, Rachel Fong, Peter Welinder, Bob McGrew, Josh Tobin, OpenAI Pieter Abbeel, and Wojciech Zaremba. Hindsight experience replay. *Advances in neural information processing systems*, 30, 2017.

Jack PK Bravo, Tyler L Dangerfield, David W Taylor, and Kenneth A Johnson. Remdesivir is a delayed translocation inhibitor of sars-cov-2 replication. *Molecular cell*, 81(7):1548–1552, 2021.

Tianle Cai, Ruiqi Gao, Jason Lee, and Qi Lei. A theory of label propagation for subpopulation shift. In *International Conference on Machine Learning*, pp. 1170–1182. PMLR, 2021.

Quanfu Fan, Yilai Li, Yuguang Yao, John Cohn, Sijia Liu, Seychelle M Vos, and Michael A Cianfrocco. Cryorl: Reinforcement learning enables efficient cryo-em data collection. *arXiv preprint arXiv:2204.07543*, 2022.

Chelsea Finn, Tianhe Yu, Justin Fu, Pieter Abbeel, and Sergey Levine. Generalizing skills with semi-supervised reinforcement learning. *arXiv preprint arXiv:1612.00429*, 2016.

Justin Fu, Katie Luo, and Sergey Levine. Learning robust rewards with adversarial inverse reinforcement learning. *arXiv preprint arXiv:1710.11248*, 2017.

Justin Fu, Avi Singh, Dibya Ghosh, Larry Yang, and Sergey Levine. Variational inverse control with events: A general framework for data-driven reward definition. *Advances in neural information processing systems*, 31, 2018.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

Jonathan Ho and Stefano Ermon. Generative adversarial imitation learning. *Advances in neural information processing systems*, 29, 2016.

Mark K Ho. The value of abstraction. *Current opinion in behavioral sciences*, 29, 2019.

Allan Jabri, Andrew Owens, and Alexei Efros. Space-time correspondence as a contrastive random walk. *Advances in neural information processing systems*, 33:19545–19560, 2020.

Nicholas K Jong and Peter Stone. State abstraction discovery from irrelevant state variables. In *IJCAI*, volume 8, pp. 752–757. Citeseer, 2005.

Zunlong Ke, Joaquin Oton, Kun Qu, Mirko Cortese, Vojtech Zila, Lesley McKeane, Takanori Nakane, Jasenko Zivanov, Christopher J Neufeldt, Berati Cerikan, et al. Structures and distributions of sars-cov-2 spike proteins on intact virions. *Nature*, 588(7838):498–502, 2020.

Goran Kokic, Hauke S Hillen, Dimitry Tegunov, Christian Dienemann, Florian Seitz, Jana Schmitzova, Lucas Farnung, Aaron Siewert, Claudia Höbartner, and Patrick Cramer. Mechanism of sars-cov-2 polymerase stalling by remdesivir. *Nature communications*, 12(1):1–7, 2021.

Ksenia Konyushkova, Konrad Zolna, Yusuf Aytar, Alexander Novikov, Scott Reed, Serkan Cabi, and Nando de Freitas. Semi-supervised reward learning for offline reinforcement learning. *arXiv preprint arXiv:2012.06899*, 2020.

Florian A Lempp, Leah B Soriaga, Martin Montiel-Ruiz, Fabio Benigni, Julia Noack, Young-Jun Park, Siro Bianchi, Alexandra C Walls, John E Bowen, Jiayi Zhou, et al. Lectins enhance sars-cov-2 infection and influence neutralizing antibodies. *Nature*, 598(7880):342–347, 2021.

Yilai Li, Quanfu Fan, John Cohn, Veronique Demers, Ja Young Lee, Lucy Yip, Michael A Cianfrocco, and Seychelle M Vos. Optimized path planning surpasses human efficiency in cryo-em imaging. *bioRxiv*, 2022.

Antonin Raffin, Ashley Hill, René Traoré, Timothée Lesort, Natalia Díaz-Rodríguez, and David Filliat. S-rl toolbox: Environments, datasets and evaluation metrics for state representation learning. *arXiv preprint arXiv:1809.09369*, 2018.

Michael T Rosenstein, Andrew G Barto, Jennie Si, Andy Barto, Warren Powell, and Donald Wunsch. Supervised actor-critic reinforcement learning. *Learning and Approximate Dynamic Programming: Scaling Up to the Real World*, pp. 359–380, 2004.

Stephane Ross and J Andrew Bagnell. Agnostic system identification for model-based reinforcement learning. *arXiv preprint arXiv:1203.1007*, 2012.

Johannes F Scheid, Christopher O Barnes, Basak Eraslan, Andrew Hudak, Jennifer R Keeffe, Lisa A Cosimi, Eric M Brown, Frauke Muecksch, Yiska Weisblum, Shuting Zhang, et al. B cell genomics behind cross-neutralization of sars-cov-2 variants and sars-cov. *Cell*, 184(12):3205–3221, 2021.

Avi Singh, Larry Yang, Kristian Hartikainen, Chelsea Finn, and Sergey Levine. End-to-end robotic reinforcement learning without reward engineering. *arXiv preprint arXiv:1904.07854*, 2019.

Kihyuk Sohn, David Berthelot, Nicholas Carlini, Zizhao Zhang, Han Zhang, Colin A Raffel, Ekin Dogus Cubuk, Alexey Kurakin, and Chun-Liang Li. Fixmatch: Simplifying semi-supervised learning with consistency and confidence. *Advances in neural information processing systems*, 33:596–608, 2020.

Adam Stooke, Kimin Lee, Pieter Abbeel, and Michael Laskin. Decoupling representation learning from reinforcement learning. In *International Conference on Machine Learning*, pp. 9870–9879. PMLR, 2021.

Chang Su, Xiaotao Jia, Xianzhong Xie, and Yue Yu. A new random-walk based label propagation community detection algorithm. In *2015 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology (WI-IAT)*, volume 1, pp. 137–140. IEEE, 2015.

Hado Van Hasselt, Arthur Guez, and David Silver. Deep reinforcement learning with double q-learning. In *Proceedings of the AAAI conference on artificial intelligence*, volume 30, 2016.

Paul Vernaza and Manmohan Chandraker. Learning random-walk label propagation for weakly-supervised semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 7158–7166, 2017.

Daniel Wrapp, Nianshuang Wang, Kizzmekia S Corbett, Jory A Goldsmith, Ching-Lin Hsieh, Olubukola Abiona, Barney S Graham, and Jason S McLellan. Cryo-em structure of the 2019-ncov spike in the prefusion conformation. *Science*, 367(6483):1260–1263, 2020.

Hangping Yao, Yutong Song, Yong Chen, Nanping Wu, Jialu Xu, Chujie Sun, Jiaxing Zhang, Tianhao Weng, Zheyuan Zhang, Zhigang Wu, et al. Molecular architecture of the sars-cov-2 virus. *Cell*, 183(3):730–738, 2020.

Harel Yedidsion, Jennifer Suriadinata, Zifan Xu, Stefan Debruyn, and Peter Stone. A scavenger hunt for service robots. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 7774–7780. IEEE, 2021.

Wanchao Yin, Chunyou Mao, Xiaodong Luan, Dan-Dan Shen, Qingya Shen, Haixia Su, Xiaoxi Wang, Fulai Zhou, Wenfeng Zhao, Minqi Gao, et al. Structural basis for inhibition of the rna-dependent rna polymerase from sars-cov-2 by remdesivir. *Science*, 368(6498):1499–1504, 2020.

Tianhe Yu, Aviral Kumar, Yevgen Chebotar, Karol Hausman, Chelsea Finn, and Sergey Levine. How to leverage unlabeled data in offline reinforcement learning. *arXiv preprint arXiv:2202.01741*, 2022.

Xiaojin Zhu and Zoubin Ghahramani. Learning from labeled and unlabeled data with label propagation. 2002.

Xiaojin Jerry Zhu. Semi-supervised learning literature survey. 2005.