DIVERSITY FROM HUMAN FEEDBACK

Ren-Jian Wang[*], **Ke Xue**[*] **& Yutong Wang**
National Key Laboratory for Novel Software Technology, Nanjing University
School of Artificial Intelligence, Nanjing University
Nanjing, China
{wangrj,xuek,wangyt}@lamda.nju.edu.cn

Peng Yang
Southern University of Science and Technology
Shenzhen, China
yangp@sustech.edu.cn

Haobo Fu & Qiang Fu
Tencent AI Lab
Shenzhen, China
{haobofu,leonfu}@tencent.com

Chao Qian[†]
National Key Laboratory for Novel Software Technology, Nanjing University
School of Artificial Intelligence, Nanjing University
Nanjing, China
qianc@lamda.nju.edu.cn

ABSTRACT

Diversity plays a significant role in many problems, such as ensemble learning, reinforcement learning, and combinatorial optimization. How to define the diversity measure is a longstanding problem. Many methods rely on expert experience to define a proper behavior space and then obtain the diversity measure, which is, however, challenging in many scenarios. In this paper, we propose the problem of learning a behavior space from human feedback and present a general method called Diversity from Human Feedback (DivHF) to solve it. DivHF learns a behavior descriptor consistent with human preference by querying human feedback. The learned behavior descriptor can be combined with any distance measure to define a diversity measure. We demonstrate the effectiveness of DivHF by integrating it with the Quality-Diversity optimization algorithm MAP-Elites and conducting experiments on the QDax suite. The results show that DivHF learns a behavior space that aligns better with human requirements compared to direct data-driven approaches and leads to more diverse solutions under human preference. Our contributions include formulating the problem, proposing the DivHF method, and demonstrating its effectiveness through experiments.

1 INTRODUCTION

Maintaining the diversity of a set of solutions is of great importance across a wide range of scenarios, such as reinforcement learning (RL) (Lehman & Stanley, 2011; Conti et al., 2018; Eysenbach et al., 2018; Parker-Holder et al., 2020; Chalumeau et al., 2023a; Wu et al., 2023; Yao et al., 2023), ensemble learning (Brown et al., 2005; Zhou, 2012; Gomes et al., 2017; An et al., 2021; He et al., 2023), and combinatorial optimization (Do et al., 2022; Nikfarjam et al., 2022). For example, diversity is a key requirement in open-ended learning (Maley, 1999; Standish, 2003; Liu et al., 2021); in level generation tasks of RL, the generated levels should be diverse to improve the robustness of the trained policy (Jiang et al., 2021b;a; Fontaine et al., 2021; Bhatt et al., 2022; Parker-Holder et al., 2022; Zhang et al., 2023); in ensemble learning, base learners should be diverse to make the ensemble learner perform well (Tumer & Ghosh, 1995; Brown et al., 2005; Sheikh et al., 2022).

[*]Equal Contribution
[†]Corresponding Author

Diversity optimization is a general and important problem of machine learning (Brown et al., 2005). A lot of diversity optimization algorithms have been proposed to obtain a set of diverse solutions for various scenarios (Pugh et al., 2016; Eysenbach et al., 2018; Fu et al., 2023). In general, most of the algorithms can be abstracted into three steps: 1) mapping the solutions (and the corresponding data collected during the evaluation of the solutions) into a latent feature space that can reflect the features of the solutions, 2) defining the diversity measure by combining the features and a distance (or similarity) metric, 3) obtaining a set of diverse solutions under the diversity measure. Among these steps, how to define the diversity measure is a longstanding problem (Maley, 1999; Kistemaker & Whiteson, 2011; Parker-Holder et al., 2020; Grillotti & Cully, 2022a). Since there are various methods for distance measure (e.g., ℓ_2 distance and cosine similarity), the core of diversity definition lies in the definition of feature space. In this paper, we consider the application of diversity in RL and refer to the feature space as the behavior space following prior works (Pugh et al., 2016; Fontaine et al., 2020; Chalumeau et al., 2023a).

There are currently two main categories of methods for defining behavior space. One is to obtain the behavior space directly from the data themselves (Eysenbach et al., 2018; Grillotti & Cully, 2022a), i.e., totally data-driven approaches. However, without considering human requirements, the learned behavior spaces are usually far away from the ones required by humans. The other is to require an expert to define the behavior space directly (Lehman & Stanley, 2011; Mouret & Clune, 2015; Nilsson & Cully, 2021; Fontaine & Nikolaidis, 2021). However, in many scenarios, even experts find it challenging to provide an appropriate behavioral metric, e.g., Real-Time Strategy (RTS) games like StarCraft (Vinyals et al., 2017) and Multi-player Online Battle Arena (MOBA) games like Dota (Berner et al., 2019).

Despite the inability to accurately describe the behavior space within humans' minds and even provide an accurate definition, humans often have the ability to express a *preference* based on their internal behavior space, i.e., they can tell which two solutions are more similar and which two are more diverse. For instance, in cooperative games between humans and AI, human players can provide feedback on which two agent partners exhibit similar behaviors (thus making humans feel bored and uninterested) and which two demonstrate diverse behaviors (thus creating a sense of excitement and intrigue).

In this paper, we propose the problem formulation of *How to learn a behavior space from human feedback* for the first time. Then, we propose a general method Diversity from Human Feedback (DivHF) to solve the problem. DivHF learns the behavior space by querying the human preference and learning a model called behavior descriptor that is consistent with human preference. Then, we can combine the learned behavior descriptor with any distance measure to obtain the diversity measure, and use an arbitrary diversity optimization algorithm to obtain a set of diverse solutions based on it.

Our proposed method DivHF is general, which can cooperate with arbitrary diversity optimization algorithms. As an instantiation, we apply DivHF to Quality-Diversity (QD) optimization algorithms, which aim to find a diverse set of high-quality solutions of a problem and have many successful applications in RL. In particular, we conduct experiments with a representative QD algorithm MAP-Elites (ME) (Mouret & Clune, 2015; Cully et al., 2015) on the popular QDax suite (Lim et al., 2023; Chalumeau et al., 2023b). The results demonstrate that the behavior space learned by DivHF is much more consistent with human requirements than the one learned directly from the data themselves without human feedback (i.e., Auto-encoder), and the final solutions obtained by the algorithm are more diverse under human preference. The effectiveness of DivHF is further verified by illustration of the learned behavior space and hyper-parameter sensitivity analysis.

Our contributions are as follows:

- We formulate the problem of obtaining diversity from human feedback.
- We propose a general method, diversity from human feedback (DivHF), to solve the problem.
- Experimental results show that DivHF can learn the accurate behavior descriptor from human feedback and help the diversity optimization algorithms to obtain solutions that are diverse under human preference.

2 PRELIMINARIES

We focus on the diversity optimization algorithms, which can be abstracted into two parts: mapping the solutions $\{\boldsymbol{\theta}_i\}_{i=1}^N$ (and the data $\{\boldsymbol{\tau}_i\}_{i=1}^N$ collected during the evaluation of the solutions) into behavior vectors $\{\boldsymbol{d}_{\boldsymbol{\phi}}(\boldsymbol{\theta}_i, \boldsymbol{\tau}_i)\}_{i=1}^N$ by a behavior descriptor $\boldsymbol{d}_{\boldsymbol{\phi}}(\cdot, \cdot)$ parameterized by $\boldsymbol{\phi}$, and then optimizing the diversity $Div(\{\boldsymbol{d}_{\boldsymbol{\phi}}(\boldsymbol{\theta}_i, \boldsymbol{\tau}_i)\}_{i=1}^N)$ in the behavior space. For convenience, we use $\boldsymbol{x}$ to denote the pair $(\boldsymbol{\theta}, \boldsymbol{\tau})$ of a solution $\boldsymbol{\theta}$ and its corresponding evaluation data $\boldsymbol{\tau}$.

Considering the human requirements, for two pairs of solutions $(\boldsymbol{x}_1, \boldsymbol{x}_2)$ and $(\boldsymbol{x}_3, \boldsymbol{x}_4)$, we denote $(\boldsymbol{x}_1, \boldsymbol{x}_2) \succ (\boldsymbol{x}_3, \boldsymbol{x}_4)$ if and only if $(\boldsymbol{x}_1, \boldsymbol{x}_2)$ is more similar than $(\boldsymbol{x}_3, \boldsymbol{x}_4)$ under human preference.

3 RELATED WORK

Diversity Optimization. Most diversity optimization algorithms can be viewed as optimizing the diversity metrics in a latent space, e.g., behavior space. The mapping from the solutions to the behavior is the common process of the algorithms, which is critical and determines whether the algorithms can generate a set of diverse solutions that meet human requirements.

Take a popular family of diversity optimization algorithms, i.e., QD algorithms (Mouret & Clune, 2015; Cully et al., 2015; Lehman & Stanley, 2011; Chatzilygeroudis et al., 2021) as examples, which aim to generate a set of high-quality and diverse solutions. Given a fitness function f to be maximized and an expert-defined behavior descriptor $\boldsymbol{d}$, the goal of QD is to generate a set of solutions that cover the space of behavior descriptors and have high fitness values.

ME (Mouret & Clune, 2015; Cully et al., 2015), the most well-known QD algorithm, maintains an archive by discretizing the behavior space into M cells $\{\mathcal{S}_i\}_{i=1}^M$ and storing at most one solution in each cell. ME aims to fill the cells with high-quality solutions. Thus, the goal of ME is to maximize the QD-Score $\sum_{i=1}^M f(\boldsymbol{\theta}_i)$, where $\boldsymbol{\theta}_i$ represents the solution contained within the cell $\mathcal{S}_i$, i.e., $\boldsymbol{d}(\boldsymbol{\theta}_i) \in \mathcal{S}_i$. If a cell $\mathcal{S}_i$ does not contain a solution $\boldsymbol{\theta}_i$, then $f(\boldsymbol{\theta}_i)$ is considered as 0. For simplicity, the fitness value $f(\cdot)$ is assumed (or converted) to be non-negative to prevent the solutions from decreasing the QD-Score. As a type of evolutionary algorithm, the main process of ME is to select parent solutions from the archive, generate offspring solutions through variation operators, evaluate the offspring solutions, and update the archive.

Behavior Space of Diversity Optimization. According to the way the behavior space is defined, the algorithms can be usually categorized into two types. The first type is to obtain the behavior space directly from the data themselves. For example, AURORA (Cully, 2019; Grillotti & Cully, 2022a) uses auto-encoder to learn the behavior space in an unsupervised manner from raw sensory data; RUDA (Grillotti & Cully, 2022b) considers the relevance of the solutions in downstream tasks, varying the distance metric to generate more relevant solutions; Many skill discovery methods (e.g., DIAYN (Eysenbach et al., 2018) and DADS (Sharma et al., 2020)) sample several diverse behaviors from a behavior space, and train a behavior-conditioned policy to maximize the mutual information between the behaviors and the trajectories of the policy. However, as these type of methods do not consider the human requirements of diversity, the obtained diversity is usually far away from the one required by human, making them less useful in many practical and complex scenarios.

The second type considers the requirements of humans by requiring an expert to define the behavior space directly. For example, given an expert-defined behavior space, ME (Mouret & Clune, 2015; Cully et al., 2015) and NSLC (Lehman & Stanley, 2011) find a set of diverse solutions that cover the behavior space. In robotic training, in order to train a set of policies that help recover quickly from damage, the experts use the touchdown time of feet as the behavior descriptor and use ME to obtain a set of solutions with varying frequency of feet usage (Cully et al., 2015; Nilsson & Cully, 2021; Chalumeau et al., 2023a). Interactive Constrained ME (Alvarez et al., 2019; 2022) enables the human designer to flexibly change the feature dimensions (i.e., behavior space) for content generation. However, in many scenarios, it is hard to define a proper behavior space, even for the experts.

Learning from Human Feedback. Preference-based RL is proposed to provide human-preferred objectives to RL agents (Wirth et al., 2017; Christiano et al., 2017; Casper et al., 2023). In this

approach, humans are asked to provide their preferences regarding pairs of agents' historical trajectories. By using human feedback, a reward model is learned and utilized to provide learning signals to the agents. Preference-based RL offers an effective means to learn from human intentions, rather than relying solely on explicitly designed rewards. Its effectiveness has been demonstrated in various domains, such as robotics control (Zhang et al., 2019) and chatbots (Stiennon et al., 2020; Ouyang et al., 2022). Notably, reinforcement learning from human feedback (RLHF) has emerged as a key strategy for fine-tuning large language models (Open-AI, 2023; Google, 2023).

In this work, we investigate the important question for the first time: How to define a proper behavior space for diversity optimization? Note that defining a proper behavior space directly is difficult even for an expert in many scenarios. Inspired by RLHF, we propose the DivHF method to solve this problem by learning the behavior space and optimizing the diversity from human feedback. Unlike RLHF, DivHF considers finding the diversity measure rather than a reward function (or reward bonus (Hussonnois et al., 2023)). DivHF asks the human to determine which *pair of policies* is more diverse, rather than which policy is preferred. DivHF predicts the behaviors of solutions to be used by diversity optimization algorithms, rather than the rewards of state-action pairs.

4 DIVHF

4.1 PROBLEM FORMULATION

In this paper, we explore the following problem: *How to obtain a set of diverse solutions based on human preference?* This problem is common and natural since humans often find it challenging to define a proper behavior space or assign exact behavior values. Instead, it is much easier for humans to distinguish which two solutions are the most similar and which two are the most diverse from a set of solutions, which is expressed through *human feedback*. This feedback reflects their preference for the *diversity*. If we can learn a behavior descriptor to establish a behavior space, we can easily define similarity (or distance) using various measures such as cosine similarity or Euclidean distance, and obtain a set of diverse solutions based on the learned behavior descriptor by any diversity optimization algorithms. Therefore, our main objective is to efficiently address the problem of *How to learn a behavior descriptor from human feedback?*.

To be more clear, given the data $\{\boldsymbol{x}_i\}_{i=1}^N$, we ask for the human preference. If the human thinks $(\boldsymbol{x}_1, \boldsymbol{x}_2)$ is more similar than $(\boldsymbol{x}_3, \boldsymbol{x}_4)$, then we want the similarity between $\boldsymbol{d}_{\boldsymbol{\phi}}(\boldsymbol{x}_1)$ and $\boldsymbol{d}_{\boldsymbol{\phi}}(\boldsymbol{x}_2)$ to be larger than that between $\boldsymbol{d}_{\boldsymbol{\phi}}(\boldsymbol{x}_3)$ and $\boldsymbol{d}_{\boldsymbol{\phi}}(\boldsymbol{x}_4)$, to be consistent with the human preference.

4.2 OVERALL METHOD

We propose the DivHF method to learn the behavior descriptor and optimize the diversity from human feedback. The main idea is to select some data to query for human preference and train a model of behavior descriptor that is consistent with human preference. As shown in Figure 1, the main process of DivHF can be summarized as follows.

1. Select some solutions (along with the corresponding evaluation data, e.g., trajectories) as examples to query human preference (red arrows).
2. Train a behavior descriptor model to extract the behaviors of solutions according to the human preference data (yellow arrows).
3. Optimize the solutions based on the learned behavior descriptor by using an arbitrary diversity optimization algorithm (blue arrows).

4.3 DATA COLLECTION: QUERY THE HUMAN PREFERENCE

Considering the requirements of diversity optimization algorithms, a behavior descriptor model takes the solutions and/or the evaluation data as input features, and the behavior predictions as outputs. The model should be designed according to the structure of features. For example, if the evaluation data are trajectories of reinforcement learning, we can use bidirectional stacked Long Short-Term Memory (LSTM) (Hochreiter & Schmidhuber, 1997; Dyer et al., 2015) as the model.

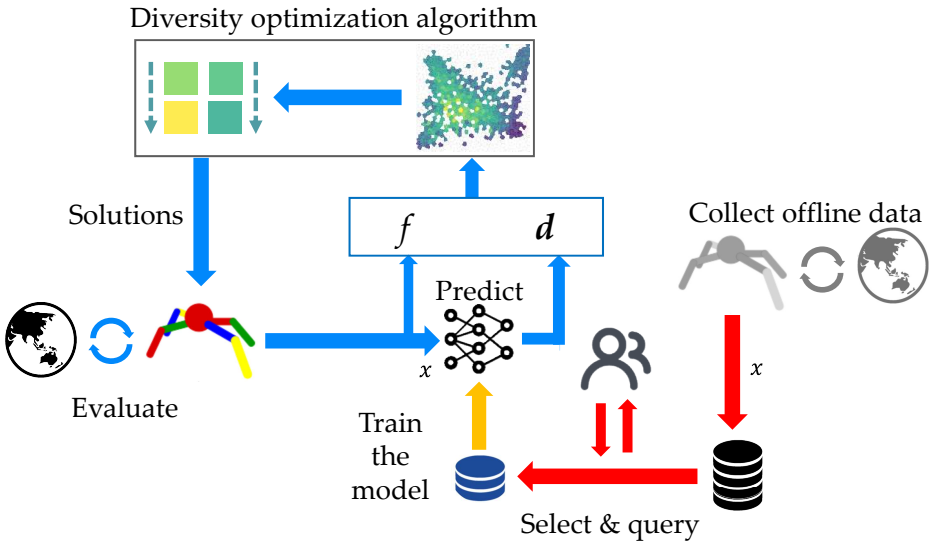

Figure 1: The workflow of DivHF.

To train the model, we need to select some solutions and their evaluation data to query human preference. DivHF takes a set of three solutions $(\boldsymbol{x}_1, \boldsymbol{x}_2, \boldsymbol{x}_3)$ as a query, and lets the human determine which two are the most similar and which two are the most diverse. If we take only two solutions as a query, the human will be required to tell their exact similarity value, which is hard for the human. If we take too many solutions as a query, it will take more time for the human to determine.

4.4 Training the Behavior Descriptor

With the human preference data, we want to train the model that can give behaviors that are consistent with human preference by a loss function and an optimizer. As shown in Figure 2, given a piece of human preference, we want to maximize the similarity of the behaviors of the most similar solutions and minimize the similarity of the behaviors of the most diverse ones according to the human preference. Given a similarity metric $\mathrm{sim}(\cdot, \cdot)$, we can predict the *pair-wise similarity* $\mathrm{sim}(\boldsymbol{d}_{\boldsymbol{\phi}}(\boldsymbol{x}_i), \boldsymbol{d}_{\boldsymbol{\phi}}(\boldsymbol{x}_j))$. Assuming, without loss of generality, that $(\boldsymbol{x}_1, \boldsymbol{x}_2)$ is the most similar and $(\boldsymbol{x}_1, \boldsymbol{x}_3)$ is the most diverse, then $\mathrm{sim}(\boldsymbol{d}_{\boldsymbol{\phi}}(\boldsymbol{x}_1), \boldsymbol{d}_{\boldsymbol{\phi}}(\boldsymbol{x}_2))$ and $\mathrm{sim}(\boldsymbol{d}_{\boldsymbol{\phi}}(\boldsymbol{x}_1), \boldsymbol{d}_{\boldsymbol{\phi}}(\boldsymbol{x}_3)))$ should be maximized and minimized, respectively. Thus, a simple loss function is to optimize the corresponding similarity metric simultaneously:

$$\mathcal{L}(\boldsymbol{\phi}) = \sum_{(\boldsymbol{x}_1, \boldsymbol{x}_2, \boldsymbol{x}_3) \in \mathcal{D}} \mathrm{sim}(\boldsymbol{d}_{\boldsymbol{\phi}}(\boldsymbol{x}_1), \boldsymbol{d}_{\boldsymbol{\phi}}(\boldsymbol{x}_3)) - \mathrm{sim}(\boldsymbol{d}_{\boldsymbol{\phi}}(\boldsymbol{x}_1), \boldsymbol{d}_{\boldsymbol{\phi}}(\boldsymbol{x}_2)). \tag{1}$$

However, since the preferences given by humans are relative, optimizing the proportion is better, which can eliminate the effects of scaling.

Another way is to use the Bradley-Terry model (Bradley & Terry, 1952), which is similar to prior works in RLHF (Christiano et al., 2017; Ibarz et al., 2018; Lee et al., 2021) and usually called InfoNCE loss (van den Oord et al., 2018). For a piece of preference data $(\boldsymbol{x}_1, \boldsymbol{x}_2, \boldsymbol{x}_3)$, we can calculate the predicted behavior $\{\boldsymbol{d}_{\boldsymbol{\phi}}(\boldsymbol{x}_i)\}_{i=1}^3$. Then, we can predict the similarities $\{\mathrm{sim}(\boldsymbol{d}_{\boldsymbol{\phi}}(\boldsymbol{x}_i), \boldsymbol{d}_{\boldsymbol{\phi}}(\boldsymbol{x}_j))\}_{i,j \in \{1,2,3\}, i \neq j}$ of the pairs $\{(\boldsymbol{x}_i, \boldsymbol{x}_j)\}_{i,j \in \{1,2,3\}, i \neq j}$. We can consider $\mathrm{sim}(\boldsymbol{d}_{\boldsymbol{\phi}}(\boldsymbol{x}_i), \boldsymbol{d}_{\boldsymbol{\phi}}(\boldsymbol{x}_j))$ as the underlying factor that reflects the human preference between $(\boldsymbol{x}_i, \boldsymbol{x}_j)$ and other pairs. We assume that the human preferences are influenced by the power of these factors. Consequently, we can forecast the probability of preference by applying the soft-max function to the respective underlying factors:

$$P_{\boldsymbol{\phi}}[(\boldsymbol{x}_i, \boldsymbol{x}_j) \succ (\boldsymbol{x}_i, \boldsymbol{x}_k)] = \frac{\exp(\lambda \mathrm{sim}(\boldsymbol{d}_{\boldsymbol{\phi}}(\boldsymbol{x}_i), \boldsymbol{d}_{\boldsymbol{\phi}}(\boldsymbol{x}_j)))}{\exp(\lambda \mathrm{sim}(\boldsymbol{d}_{\boldsymbol{\phi}}(\boldsymbol{x}_i), \boldsymbol{d}_{\boldsymbol{\phi}}(\boldsymbol{x}_j))) + \exp(\lambda \mathrm{sim}(\boldsymbol{d}_{\boldsymbol{\phi}}(\boldsymbol{x}_i), \boldsymbol{d}_{\boldsymbol{\phi}}(\boldsymbol{x}_k)))},$$

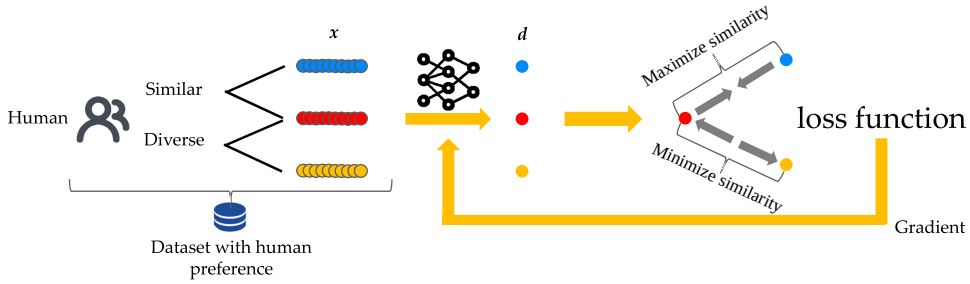

Figure 2: The training process of DivHF.

where $(\boldsymbol{x}_i, \boldsymbol{x}_j) \succ (\boldsymbol{x}_i, \boldsymbol{x}_k)$ denotes that $(\boldsymbol{x}_i, \boldsymbol{x}_j)$ is more similar than $(\boldsymbol{x}_i, \boldsymbol{x}_k)$, and λ is the temperature that adjusts the discrimination between positive and negative samples. The goal is to learn $\boldsymbol{d}_{\boldsymbol{\phi}}$ that minimizes the cross-entropy between the predictions and the human preferences:

$$\mathcal{L}(\boldsymbol{\phi}) = - \sum_{(\boldsymbol{x}_1, \boldsymbol{x}_2, \boldsymbol{x}_3) \in \mathcal{D}} \log P_{\boldsymbol{\phi}}[(\boldsymbol{x}_1, \boldsymbol{x}_2) \succ (\boldsymbol{x}_1, \boldsymbol{x}_3)], \tag{2}$$

where we assume, without loss of generality, that $(\boldsymbol{x}_1, \boldsymbol{x}_2)$ is the most similar and $(\boldsymbol{x}_1, \boldsymbol{x}_3)$ is the most diverse. Note that we only use the most similar pair and the most diverse pair, as they are easier to determine by humans.

5 Experiments

To examine the performance of DivHF, we conduct experiments on the popular QDax domain (Lim et al., 2023; Chalumeau et al., 2023b), including HalfCheetah, Walker2D, Ant, and Humanoid. The goal of the tasks is to generate a set of robotic policies that are diverse in the frequency of the usage of each foot, in order to enable the robot to recover quickly from damage. We collect 25k trajectories offline to train the behavior descriptor model. For convenience, we define an oracle behavior space for each task, i.e., the fraction of time each foot touches the ground. We also use a synthetic oracle whose preferences exactly reflect the similarity of the oracle behaviors. That is, when the method queries for a preference, we immediately reply which two trajectories have the most similar oracle behaviors and which two have the most diverse ones.

In our experiments, we use a bidirectional two-layer-stacked LSTM (Hochreiter & Schmidhuber, 1997; Dyer et al., 2015) followed by a two-layer MLP to extract the feature from the temporal trajectory data, where the feature sizes of LSTM layers are all 512, and the hidden layer size of MLP layers is 32. We use the cosine similarity as the similarity metric. The dimension of the learned space is the same as the oracle behavior space.

To evaluate the effectiveness of DivHF, we compare the following methods. **DivHF**: The proposed method that uses cross-entropy in Eq. (2) as the loss function. **DivHF w/o CE**: The same as DivHF, except that it uses simple loss in Eq. (1) as the loss function. **DivHF w/o BT**: The same as DivHF, except that it uses simple LSTM as the model. **DivHF-Vanilla**: The same as DivHF, except that it uses simple loss in Eq. (1) as the loss function and simple LSTM as the model. **Auto-encoder**: The method that concatenates a decoder after the behavior descriptor model and learns the behavior space directly by self-supervision without human preference. **Oracle**: The oracle behavior descriptor itself.

After training, all the behavior description methods are in conjunction with ME (Mouret & Clune, 2015; Cully et al., 2015) to generate a set of diverse solutions. Specifically, we use the learned mapping instead of the expert-defined behavior mapping to obtain the behaviors of the solutions in the ME framework.

The experiments aim to answer the following two Research Questions (RQs): 1) Whether the learned behavior descriptors are consistent with human preferences? 2) When applied to diversity optimization algorithms (e.g., QD algorithms), whether the solutions are diverse under human preference?

5.1 RQ1: Whether the Learned Behavior Descriptors are Consistent with Human Preferences?

We first train the models with the human preference data and use the following metrics to evaluate whether the behavior descriptors learned by different methods are consistent with human preference.

- **Most Similar Accuracy**: The accuracy of selecting the most similar pair in each triple of data, which is calculated by the proportion of making correct selections.
- **Most Diverse Accuracy**: The accuracy of selecting the most diverse pair in each triple of data.
- **Preference Accuracy**: The accuracy of selecting the most similar pair and the most diverse pair in each triple of data at a time. Note that as there are three pairs in each triple of data, selecting the most similar pair and the most diverse pair correctly also implies predicting the preference of three pairs correctly.
- **Pair-wise Accuracy**: The accuracy of predicting the preference of two pairs, i.e., whether one pair is more diverse than the other under human preference, in each triple of data.

The results are shown in Figure 3. The accuracy of all variants of DivHF is better than Auto-encoder in all environments, indicating that DivHF can really learn from human preferences and predict behaviors that are consistent with human preferences. DivHF achieves the best accuracy in most cases, except the environment Humanoid Uni where DivHF w/o CE (i.e., DivHF using simple loss in Eq. (1)) performs better. This may be due to the challenging nature of Humanoid Uni, which has the largest search space among these four environments and leads to the worst performance of all the DivHF variants. Overall, DivHF has the best average ranking, as shown in Table 1.

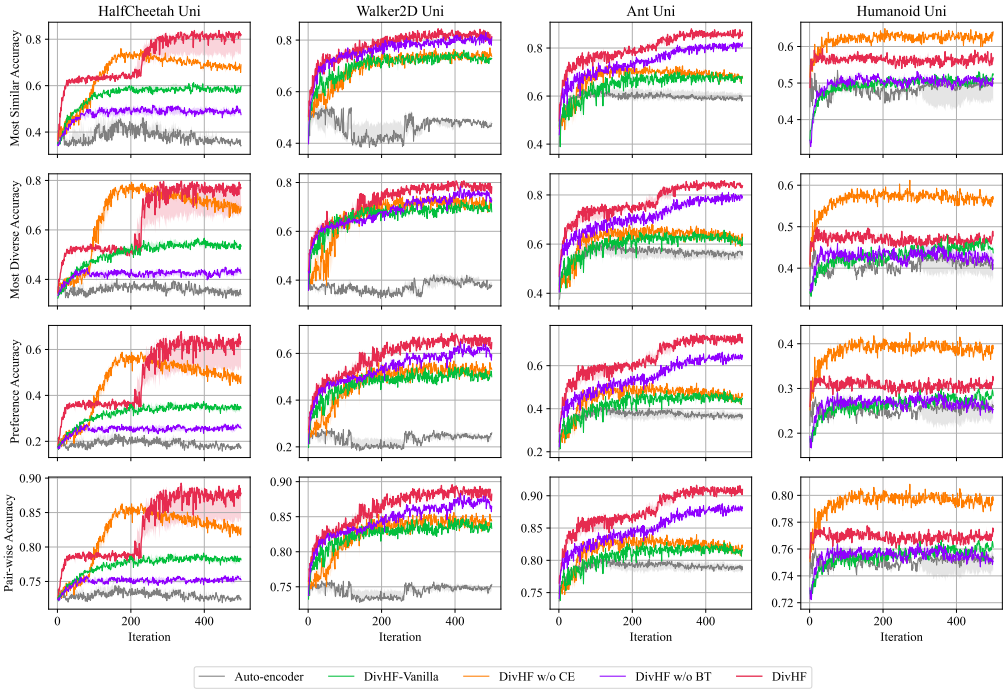

Figure 3: Accuracy of different methods on the test sets under different environments. The medians and the first and third quartile intervals are depicted with curves and shaded areas, respectively.

5.2 RQ2: When Applied to Classical Algorithms, whether the Solutions are Diverse under Human Preference?

Next, we use the learned behavior models as the behavior descriptors of classical QD algorithms and examine whether they can obtain a set of solutions that are diverse under human preference. In

Environment	Auto-encoder	DivHF-Vanilla	DivHF w/o CE	DivHF w/o BT	DivHF
HalfCheetah Uni	0.171	0.347	0.487	0.265	**0.563**
Walker2D Uni	0.246	0.501	0.521	0.594	**0.655**
Ant Uni	0.365	0.453	0.444	0.644	**0.719**
Humanoid Uni	0.251	0.288	**0.395**	0.261	0.312
Average Ranking	5.00	3.25	2.50	3.00	**1.25**

Table 1: Preference Accuracy comparisons (mean). **Bold** texts denote the best method.

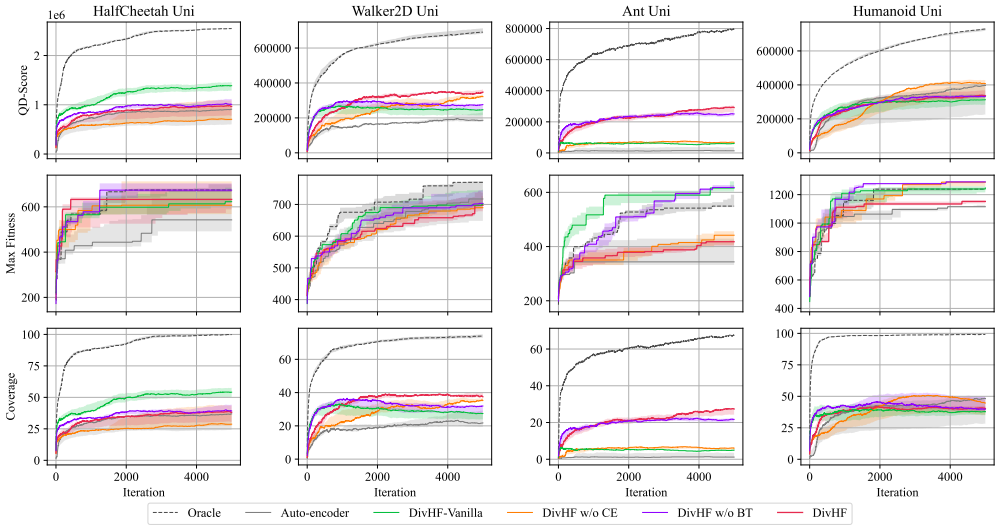

Figure 4: Performance comparisons in terms of QD-Score, Coverage, and Max Fitness under the four environments. The medians and the first and third quartile intervals are depicted with curves and shaded areas, respectively.

order to examine the diversity, after each generation, we map the solutions to the oracle behavior space and put them into an oracle CVT archive. Then, we calculate the classical QD metrics to show the diversity under human preference.

The following QD metrics are considered. 1) **QD-Score**: The total sum of the fitness values across all solutions in the oracle CVT archive. It can measure both the quality and the diversity of the solutions. 2) **Max Fitness**: The largest fitness value of the solutions in the archive. It can measure the quality of the solutions. 3) **Coverage**: The total number of solutions in the oracle CVT archive. It can measure the diversity of the solutions.

The results are shown in Figure 4. The QD-Score and Coverage are more related to the goodness of the learned behavior. In general, the higher the accuracy of the learned behavior, the higher QD-Score obtained. As shown in Table 1 and 2, the average rankings of the accuracy of behavior and QD-Score are consistent: Oracle $<$ DivHF $<$ DivHF w/o CE $<$ DivHF w/o BT $<$ DivHF-Vanilla $<$ Auto-encoder. We can also observe from Figure 4 that compared with the oracle behavior descriptor, the QD-Score and Coverage of DivHF still have a large gap, implying that further improvement is expected in the future.

Furthermore, we visualize the solutions in the oracle behavior space and the learned behavior space. As shown in Figure 5, the sub-figures (a) and (c) denote the oracle behavior space, and the sub-figures (b) and (d) denote the learned behavior space. The color in sub-figures (a) and (b) denote the quality of the solutions. In sub-figure (c), we use different colors (black, red, pink, and purple) to color the oracle behavior space. Then, we map the solutions with their colors to the learned behavior space, i.e., sub-figure (d). Sub-figures (c) and (d) show a clear resemblance between the oracle space and the learned behavior space: the oracle behavior space can be recovered roughly by rotating the

Environment	Auto-encoder	DivHF-Vanilla	DivHF w/o CE	DivHF w/o BT	DivHF	Oracle
HalfCheetah Uni	749,067	**1,368,209**	946,628	1,070,816	852,532	2,550,238
Walker2D Uni	209,583	227,111	324,036	258,224	**353,520**	700,442
Ant Uni	26,170	61,090	61,967	247,828	**278,938**	796,666
Humanoid Uni	289,382	313,680	**363,041**	334,443	349,922	725,198
Average Ranking	6.00	4.25	3.25	3.50	**3.00**	1.00

Table 2: QD-Score comparisons (mean). **Bold** texts denote the best method except for Oracle.

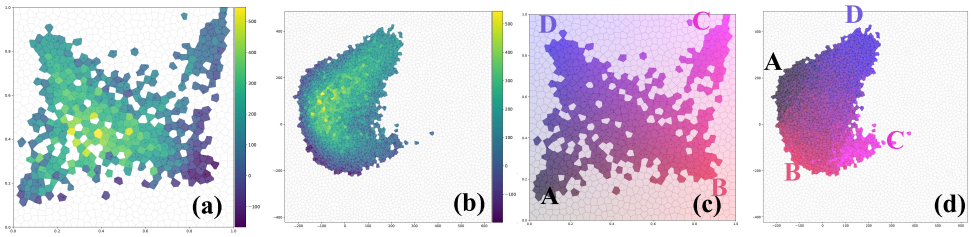

Figure 5: Visualization of the solutions obtained from DivHF. The sub-figures (a) and (c) denote the oracle behavior space, and the sub-figures (b) and (d) denote the learned behavior space. The color in sub-figures (a) and (b) denote the quality of the solutions. We put gradient colors on sub-figure (c), and map the solutions with their colors to sub-figure (d).

learned behavior space counterclockwise. This indicates that the learned behavior space captures the essence of the oracle behavior space.

5.3 Additional Studies

Dimension of Behavior Space. We examine the influence of the dimension of the behavior space. We compare the setting of using the same dimension as the oracle behavior and that of using a doubled dimension, which are denoted as DivHF and DivHF ($2\times$dim), respectively. Figure 6 shows that a higher dimension can lead to better accuracy due to the better representation ability of the model. However, a behavior space with too high dimension may be hard for humans to understand, and also hard for diversity optimization algorithms to optimize.

Number of Queries. We further examine the influence of the number of queries used for training. We compare the settings that use 10k, 25k, and 50k queries. As expected, Figure 7 shows that using a too small number of queries may lead to poor performance. The use of many human queries is one of the limitations of the proposed method, and we will consider reducing the number of human queries in future work.

6 Discussion

Diversity plays an important role in many scenarios and usually relies on a well-defined behavior descriptor, which is, however, challenging in practice. In this paper, we introduce a natural way of defining diversity from human feedback for the first time. We propose a general method called DivHF, which selects triples of data to query humans about which two are the most similar and which two are the most diverse, and trains the behavior descriptor with cross-entropy loss to make it consistent with human preference. DivHF can cooperate with arbitrary diversity optimization algorithms, and we apply it to QD algorithms as an instantiation. Experimental results on QDax show that DivHF can learn the accurate behavior descriptor from human feedback.

This paper is a preliminary work that obtains diversity from human feedback. There are many interesting future works, including improving the accuracy of the models, reducing the number of human queries, applying to other diversity optimization algorithms and other scenarios, and handling the diverse underlying preferences of different humans effectively.

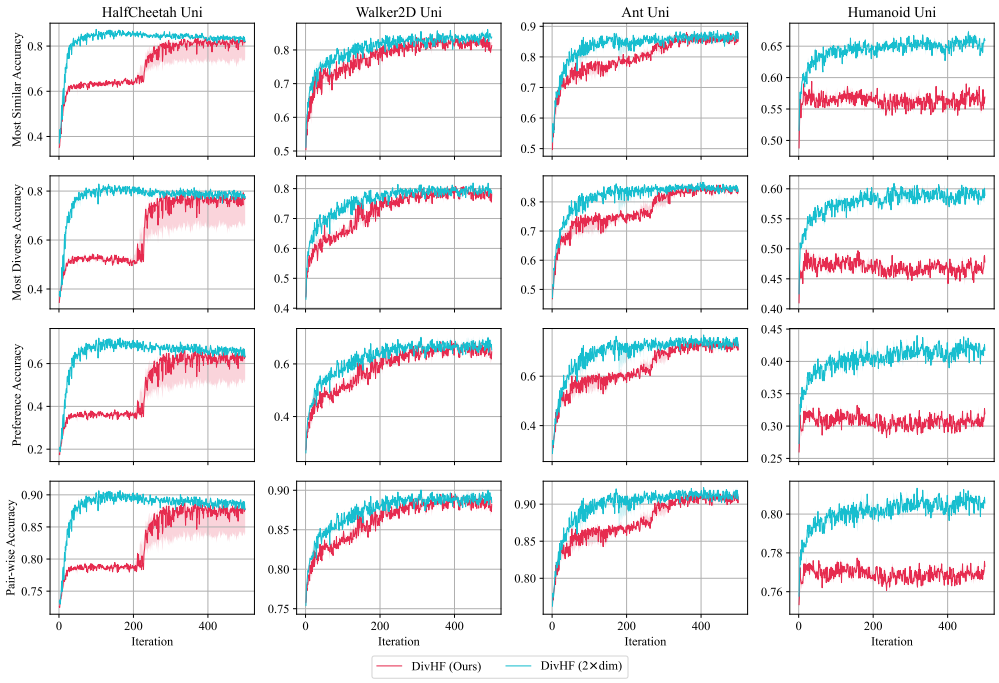

Figure 6: Accuracy of DivHF with different dimensions of behavior descriptor, where DivHF and DivHF (2×dim) denote that the dimension of the learned behavior descriptor is the same as and twice of the dimension of the oracle behavior descriptor, respectively. The medians and the first and third quartile intervals are depicted with curves and shaded areas, respectively.

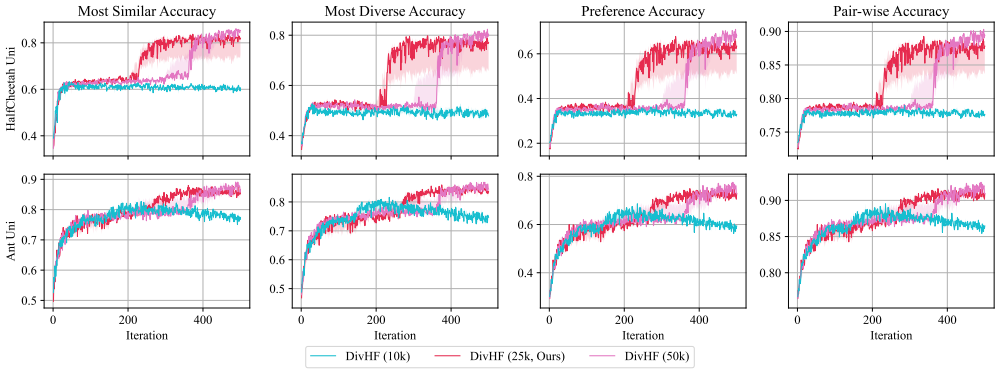

Figure 7: Accuracy of DivHF using different sizes of training set, i.e., different number of queries in training. The medians and the first and third quartile intervals are depicted with curves and shaded areas, respectively. 25k is used in our experiments.

REFERENCES

Alberto Alvarez, Steve Dahlskog, Jose Font, and Julian Togelius. Empowering quality diversity in dungeon design with interactive constrained MAP-Elites. In *Proceedings of the IEEE Conference on Games (CoG)*, pp. 1–8, London, United Kingdom, 2019.

Alberto Alvarez, Steve Dahlskog, Jose Font, and Julian Togelius. Interactive constrained MAP-Elites: Analysis and evaluation of the expressiveness of the feature dimensions. *IEEE Transactions on Games*, 14(2):202–211, 2022.

Gaon An, Seungyong Moon, Jang-Hyun Kim, and Hyun Oh Song. Uncertainty-based offline reinforcement learning with diversified Q-ensemble. In *Advances in Neural Information Processing*

Systems 34 (NeurIPS), pp. 7436–7447, Virtual, 2021.

Christopher Berner, Greg Brockman, Brooke Chan, Vicki Cheung, Przemysław Dkebiak, Christy Dennison, David Farhi, Quirin Fischer, Shariq Hashme, Chris Hesse, et al. Dota 2 with large scale deep reinforcement learning. *CoRR abs/1912.06680*, 2019.

Varun Bhatt, Bryon Tjanaka, Matthew C. Fontaine, and Stefanos Nikolaidis. Deep surrogate assisted generation of environments. In *Advances in Neural Information Processing Systems 35 (NeurIPS)*, New Orleans, LA, 2022.

Ralph Allan Bradley and Milton E Terry. Rank analysis of incomplete block designs: I. The method of paired comparisons. *Biometrika*, 39(3/4):324–345, 1952.

Gavin Brown, Jeremy Wyatt, Rachel Harris, and Xin Yao. Diversity creation methods: A survey and categorisation. *Information Fusion*, 6(1):5–20, 2005.

Stephen Casper, Xander Davies, Claudia Shi, Thomas Krendl Gilbert, Jérémy Scheurer, Javier Rando, Rachel Freedman, Tomasz Korbak, David Lindner, Pedro Freire, Tony 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. Open problems and fundamental limitations of reinforcement learning from human feedback. *CoRR*, abs/2307.15217, 2023.

Felix Chalumeau, Raphael Boige, Bryan Lim, Valentin Macé, Maxime Allard, Arthur Flajolet, Antoine Cully, and Thomas Pierrot. Neuroevolution is a competitive alternative to reinforcement learning for skill discovery. In *The 11th International Conference on Learning Representations (ICLR)*, Kigali, Rwanda, 2023a.

Felix Chalumeau, Bryan Lim, Raphael Boige, Maxime Allard, Luca Grillotti, Manon Flageat, Valentin Macé, Arthur Flajolet, Thomas Pierrot, and Antoine Cully. QDax: A library for quality-diversity and population-based algorithms with hardware acceleration. *arXiv:2308.03665*, 2023b.

Konstantinos Chatzilygeroudis, Antoine Cully, Vassilis Vassiliades, and Jean-Baptiste Mouret. Quality-diversity optimization: A novel branch of stochastic optimization. In *Black Box Optimization, Machine Learning, and No-Free Lunch Theorems*, pp. 109–135. Springer, 2021.

Paul F Christiano, Jan Leike, Tom Brown, Miljan Martic, Shane Legg, and Dario Amodei. Deep reinforcement learning from human preferences. In *Advances in Neural Information Processing Systems 30 (NIPS)*, pp. 4299–4307, Long Beach, CA, 2017.

Edoardo Conti, Vashisht Madhavan, Felipe P. Such, J. Lehman, Kenneth O. Stanley, and Jeff Clune. Improving exploration in evolution strategies for deep reinforcement learning via a population of novelty-seeking agents. In *Advances in Neural Information Processing Systems 32 (NeurIPS)*, pp. 5032–5043, Montréal, Canada, 2018.

Antoine Cully. Autonomous skill discovery with quality-diversity and unsupervised descriptors. In *Proceedings of the 21th Genetic and Evolutionary Computation Conference (GECCO)*, pp. 81–89, Lille, France, 2019.

Antoine Cully, Jeff Clune, Danesh Tarapore, and Jean-Baptiste Mouret. Robots that can adapt like animals. *Nature*, 521(7553):503–507, 2015.

Anh Do, Mingyu Guo, Aneta Neumann, and Frank Neumann. Analysis of evolutionary diversity optimization for permutation problems. *ACM Transactions on Evolutionary Learning and Optimization*, 2(3):1–27, 2022.

Chris Dyer, Miguel Ballesteros, Wang Ling, Austin Matthews, and Noah A. Smith. Transition-based dependency parsing with stack long short-term memory. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers) (ACL)*, pp. 334—-343, Beijing, China, 2015.

Benjamin Eysenbach, Abhishek Gupta, Julian Ibarz, and Sergey Levine. Diversity is all you need: Learning skills without a reward function. In *The 6th International Conference on Learning Representations (ICLR)*, Vancouver, Canada, 2018.

Matthew Fontaine and Stefanos Nikolaidis. Differentiable quality diversity. In *Advances in Neural Information Processing Systems 34 (NeurIPS)*, pp. 10040–10052, Virtual, 2021.

Matthew C. Fontaine, Julian Togelius, Stefanos Nikolaidis, and Amy K. Hoover. Covariance matrix adaptation for the rapid illumination of behavior space. In *Proceedings of the 22th ACM Genetic and Evolutionary Computation Conference (GECCO)*, pp. 94–102, Cancún, Mexico, 2020.

Matthew C. Fontaine, Ruilin Liu, Ahmed Khalifa, Jignesh Modi, Julian Togelius, Amy K. Hoover, and Stefanos Nikolaidis. Illuminating mario scenes in the latent space of a generative adversarial network. In *Proceedings of the 35th AAAI Conference on Artificial Intelligence (AAAI)*, pp. 5922–5930, Virtual, 2021.

Wei Fu, Weihua Du, Jingwei Li, Sunli Chen, Jingzhao Zhang, and Yi Wu. Iteratively learn diverse strategies with state distance information. In *Advances in Neural Information Processing Systems 36 (NeurIPS)*, New Orleans, LA, 2023.

Heitor Murilo Gomes, Jean Paul Barddal, Fabrício Enembreck, and Albert Bifet. A survey on ensemble learning for data stream classification. *ACM Computing Surveys*, 50(2):1–36, 2017.

Google. Bard. 2023. URL `https://bard.google.com/`.

Luca Grillotti and Antoine Cully. Unsupervised behavior discovery with quality-diversity optimization. *IEEE Transactions on Evolutionary Computation*, 26(6):1539–1552, 2022a.

Luca Grillotti and Antoine Cully. Relevance-guided unsupervised discovery of abilities with quality-diversity algorithms. In *Proceedings of the 24th ACM Genetic and Evolutionary Computation Conference (GECCO)*, pp. 77—-85, Boston, MA, 2022b.

Yi-Xiao He, Yu-Chang Wu, Chao Qian, and Zhi-Hua Zhou. Margin distribution and structural diversity guided ensemble pruning. *Machine Learning*, 2023.

Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural Computation*, 9(8): 1735–1780, 1997.

Maxence Hussonnois, Thommen George Karimpanal, and Santu Rana. Controlled diversity with preference: Towards learning a diverse set of desired skills. In *Proceedings of the International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, pp. 1135–1143, London, United Kingdom, 2023.

Borja Ibarz, Jan Leike, Tobias Pohlen, Geoffrey Irving, Shane Legg, and Dario Amodei. Reward learning from human preferences and demonstrations in atari. In *Advances in Neural Information Processing Systems 31 (NeurIPS)*, Montréal, Canada, 2018.

Minqi Jiang, Michael D. Dennis, Jack Parker-Holder, Jakob Nicolaus Foerster, Edward Grefenstette, and Tim Rocktäschel. Replay-guided adversarial environment design. In *Advances in Neural Information Processing Systems 34 (NeurIPS)*, Virtual, 2021a.

Minqi Jiang, Edward Grefenstette, and Tim Rocktäschel. Prioritized level replay. In *Proceedings of the 38th International Conference on Machine Learning (ICML)*, pp. 4940–4950, Virtual, 2021b.

Steijn Kistemaker and Shimon Whiteson. Critical factors in the performance of novelty search. In *Proceedings of the 13th annual conference on Genetic and evolutionary computation (GECCO)*, pp. 965–972, Dublin, Ireland, 2011.

Kimin Lee, Laura M Smith, and Pieter Abbeel. Pebble: Feedback-efficient interactive reinforcement learning via relabeling experience and unsupervised pre-training. In *Proceedings of the 38th International Conference on Machine Learning (ICML)*, pp. 6152–6163, Virtual, 2021.

Joel Lehman and Kenneth O. Stanley. Evolving a diversity of virtual creatures through novelty search and local competition. In *Proceedings of the 13th ACM Genetic and Evolutionary Computation Conference (GECCO)*, pp. 211–218, Dublin, Ireland, 2011.

Bryan Lim, Maxime Allard, Luca Grillotti, and Antoine Cully. Accelerated quality-diversity through massive parallelism. *Transactions on Machine Learning Research*, 2023.

Xiangyu Liu, Hangtian Jia, Ying Wen, Yujing Hu, Yingfeng Chen, Changjie Fan, Zhipeng Hu, and Yaodong Yang. Towards unifying behavioral and response diversity for open-ended learning in zero-sum games. In *Advances in Neural Information Processing Systems 34 (NeurIPS)*, pp. 941–952, 2021.

Carlo C Maley. Four steps toward open-ended evolution. In *Proceedings of the 1st Annual Conference on Genetic and Evolutionary Computation (GECCO)*, pp. 1336–1343, 1999.

Jean-Baptiste Mouret and Jeff Clune. Illuminating search spaces by mapping elites. *arXiv:1504.04909*, 2015.

Adel Nikfarjam, Amirhossein Moosavi, Aneta Neumann, and Frank Neumann. Computing high-quality solutions for the patient admission scheduling problem using evolutionary diversity optimisation. In *Proceedings of the 17th International Conference on Parallel Problem Solving from Nature (PPSN)*, pp. 250–264, Dortmund, Germany, 2022.

Olle Nilsson and Antoine Cully. Policy gradient assisted MAP-Elites. In *Proceedings of the 23th ACM Genetic and Evolutionary Computation Conference (GECCO)*, pp. 866–875, Lille, France, 2021.

Open-AI. GPT-4 technical report. 2023. URL `https://openai.com/research/gpt-4`.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul F. Christiano, Jan Leike, and Ryan Lowe. Training language models to follow instructions with human feedback. In *Advances in Neural Information Processing Systems 35 (NeurIPS)*, New Orleans, LA, 2022.

Jack Parker-Holder, Aldo Pacchiano, Krzysztof M. Choromanski, and Stephen J. Roberts. Effective diversity in population based reinforcement learning. In *Advances in Neural Information Processing Systems 34 (NeurIPS)*, pp. 18050–18062, Virtual, 2020.

Jack Parker-Holder, Minqi Jiang, Michael Dennis, Mikayel Samvelyan, Jakob Foerster, Edward Grefenstette, and Tim Rocktäschel. Evolving curricula with regret-based environment design. In *Proceedings of the 39th International Conference on Machine Learning (ICML)*, pp. 17473–17498, Baltimore, MD, 2022.

Justin K. Pugh, Lisa B. Soros, and Kenneth O. Stanley. Quality Diversity: A new frontier for evolutionary computation. *Frontiers Robotics AI*, 3:40, 2016.

Archit Sharma, Shixiang Gu, Sergey Levine, Vikash Kumar, and Karol Hausman. Dynamics-aware unsupervised discovery of skills. In *The 8th International Conference on Learning Representations (ICLR)*, Virtual, 2020.

Hassam Sheikh, Kizza Frisbee, and Mariano Phielipp. DNS: Determinantal point process based neural network sampler for ensemble reinforcement learning. In *Proceedings of the 39th International Conference on Machine Learning (ICML)*, pp. 19731–19746, Baltimore, MD, 2022.

Russell K. Standish. Open-ended artificial evolution. *Int. J. Comput. Intell. Appl.*, 3(2):167–175, 2003. doi: 10.1142/S1469026803000914. URL `https://doi.org/10.1142/S1469026803000914`.

Nisan Stiennon, Long Ouyang, Jeffrey Wu, Daniel M. Ziegler, Ryan Lowe, Chelsea Voss, Alec Radford, Dario Amodei, and Paul F. Christiano. Learning to summarize with human feedback. In Hugo Larochelle, Marc'Aurelio Ranzato, Raia Hadsell, Maria-Florina Balcan, and Hsuan-Tien Lin (eds.), *Advances in Neural Information Processing Systems 33 (NeurIPS)*, Virtual, 2020.

Kagan Tumer and Joydeep Ghosh. Theoretical foundations of linear and order statistics combiners for neural pattern classifiers. *IEEE Transactions on Neural Networks*, 1995.

Aäron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv:1807.03748*, 2018.

O. Vinyals, T. Ewalds, S. Bartunov, P. Georgiev, A. S. Vezhnevets, M. Yeo, A. Makhzani, H. Küttler, J. Agapiou, J. Schrittwieser, et al. Starcraft ii: A new challenge for reinforcement learning. *CoRR abs/1708.04782*, 2017.

Christian Wirth, Riad Akrour, Gerhard Neumann, and Johannes Fürnkranz. A survey of preference-based reinforcement learning methods. *J. Mach. Learn. Res.*, 18:136:1–136:46, 2017.

Shuang Wu, Jian Yao, Haobo Fu, Ye Tian, Chao Qian, Yaodong Yang, Qiang Fu, and Yang Wei. Quality-similar diversity via population based reinforcement learning. In *The 11th International Conference on Learning Representations (ICLR)*, 2023.

Jian Yao, Weiming Liu, Haobo Fu, Yaodong Yang, Stephen McAleer, Qiang Fu, and Wei Yang. Policy space diversity for non-transitive games, 2023.

Marvin Zhang, Sharad Vikram, Laura M. Smith, Pieter Abbeel, Matthew J. Johnson, and Sergey Levine. SOLAR: deep structured representations for model-based reinforcement learning. In *Proceedings of the 36th International Conference on Machine Learning (ICML)*, pp. 7444–7453, Long Beach, CA, 2019.

Yulun Zhang, Matthew C. Fontaine, Varun Bhatt, Stefanos Nikolaidis, and Jiaoyang Li. Multi-robot coordination and layout design for automated warehousing. In *Proceedings of the 32nd International Joint Conference on Artificial Intelligence (IJCAI)*, pp. 5503–5511, Macao, SAR, China, 2023.

Zhi-Hua Zhou. *Ensemble Methods: Foundations and Algorithms*. CRC Press, 2012.