Diversifying Training Pool Predictability for Zero-shot Coordination: A Theory of Mind Approach

Dung Nguyen , Hung Le , Kien Do , Sunil Gupta , Svetha Venkatesh , Truyen Tran
Applied Artificial Intelligence Institute (A^2I^2), Deakin University
{dung.nguyen,thai.le,k.do,sunil.gupta,svetha.venkatesh,truyen.tran}@deakin.edu.au

Abstract

The challenge in constructing artificial social agents is to enable the ability to adapt to novel agents, and is called zero-shot coordination (ZSC). A promising approach is to train the adaptive agents by interacting with a diverse pool of collaborators, assuming that the greater the diversity in other agents seen during training, the better the generalisation. In this paper, we explore an alternative procedure by considering the behavioural predictability of collaborators, i.e. whether their actions and intentions are predictable, and use it to select a diverse set of agents for the training pool. More specifically, we develop a pool of agents through self-play training during which agents' behaviour evolves and has diversity in levels of behavioural predictability (LoBP) through its evolution. We construct an observer to compute the level of behavioural predictability for each version of the collaborators. To do so, the observer is equipped with the theory of mind (ToM) capability to learn to infer the actions and intentions of others. We then use an episodic memory based on the LoBP metric to maintain agents with different levels of behavioural predictability in the pool of agents. Since behaviours that emerge at the later training phase are more complex and meaningful, the memory is updated with the latest versions of training agents. Our extensive experiments demonstrate that LoBP-based diversity training leads to better ZSC than other diversity training methods.

1 Introduction

Building social agents that can coordinate with novel agents is challenging [Stone *et al.*, 2010; Dafoe *et al.*, 2021; Mirsky *et al.*, 2022; Treutlein *et al.*, 2021; Bowling and McCracken, 2005; Bard *et al.*, 2020] because artificial social agents need to adapt to diverse partners. To overcome this challenge, recent works generate a pool of *diverse* artificial agents and train the adaptive agent to cooperate with agents from this pool. This kind of generative framework to create high-quality training data [Jiang *et al.*, 2023] is actively studied in deep learning [Zha *et al.*, 2023b; Whang *et al.*, 2023; Zha *et al.*, 2023a; Jakubik *et al.*, 2022; Mazumder *et al.*, 2022] and benefits different sub-fields such as computer vision [Paul *et al.*, 2021] and natural language processing [Yu *et al.*, 2022b; Brown *et al.*, 2020]. In the context of building adaptive agents, manipulating the process of generating the training policies provides the advantage of instilling different prior knowledge and inducing flexibility in cooperating with the partner's behaviour in the training pool. To create a pool of *diverse* agents, existing works have studied various schemes to generate pools that contain agents with varying trajectories [Lupu *et al.*, 2021; Lou *et al.*, 2023; Zhao *et al.*, 2023], skills [Szot *et al.*, 2023], or desires [Yu *et al.*, 2022a].

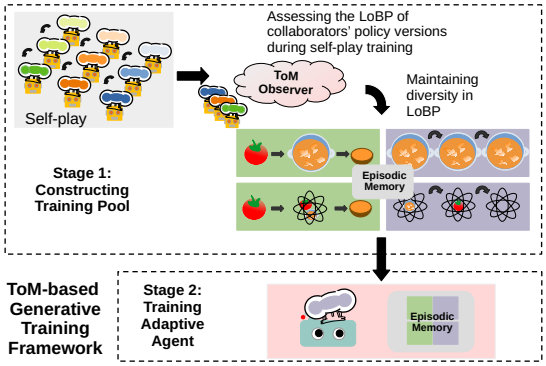

Figure 1: ToM based two-stage framework for training the adaptive agent in ZSC. In the *first* stage, a pool of collaborators is constructed via SP training. A ToM observer learns to predict the behaviours and assigns the LoBP. Based on this, an episodic memory is updated so that it maintains the diversity in the LoBP of the training pool. It also selects meaningful behaviours emerged in the later stage (green squares), instead of poor behaviours presented in the initial stage (purple squares). In the *second* stage, the adaptive agent is trained to cooperate with collaborators stored in the episodic memory.

In this paper, we explore a new diversity scheme based on the predictability of others in the team in training adaptive agents in a zero-shot coordination setting. If an agent is only trained with collaborators whose behaviours have high predictability, it will develop specific habits to cooperate with this population and not readjust to novel partners, i.e. will struggle when its new partners are less predictable. Negative effects also can happen when the adaptive agent is only

trained with agents that are completely random in behaviour, i.e. low in behavioural predictability, and bring no benefit to the team since this can potentially lead the adaptive agent to learn to achieve the task individually without cooperating with others. Hence, we suggest maintaining agents with different *levels of behavioural predictability (LoBP)* in the population with which we train the adaptive agent. An agent or a policy is unpredictable if its behaviour cannot be predicted from a third-person point of view—the theory of mind observer—, and vice versa[1]. We continually train a *theory of mind observer* to predict the behaviour, including one-step-ahead primitive actions and intentions, of self-play agents to assess the behavioural predictability of agents in a particular environment. Although recent research has considered machine theory of mind as a crucial component for artificial agents in cooperative multi-agent settings [Rabinowitz *et al.*, 2018; Papoudakis *et al.*, 2021], using this model to maintain the diversity of agents has not been studied.

Our training framework is presented in Fig. 1. In the first stage, a training pool of diverse agents is developed via self-play reinforcement learning (RL), and we pick the agents for a training pool for the next stage based on LoBP-based selection criteria. In the second stage, the adaptive agent is trained to cooperate with the agents in this training pool. We focus on the first stage, where the training pool is generated to have diverse LoBP. Our proposed approach includes (1) a *theory of mind observer*, which assigns a behavioural predictability score for agents in the pool; and (2) *an episodic memory*, which supports the construction of the training pool with diverse LoBP and favours agents that have meaningful behaviours that appeared in the later stage of training. We conducted experiments on the Overcooked environment [Carroll *et al.*, 2019] and demonstrated that the LoBP-based mechanism of selecting a training pool helps the adaptive agent learn to cooperate better with unseen partners compared to other diversity-based baselines.

To summarise, our contributions are:

- A generative training framework with a theory of mind observer and episodic memory that employs the behavioural predictability to create a pool of diverse agents and is helpful to train the adaptive agent;
- Empirically showing our method produced the adaptive agents that outperform state-of-the-art methods on different cooperative tasks when paired with novel partners;
- An in-depth analysis of the pool of agents developed during the selection process.

2 Related Works

Adaptability of reinforcement learning agents in social setting and zero-shot coordination. Creating a pool of agents and training an adaptive agent to best respond to this pool is emerging research to achieve zero-shot coordination. Approaches following these lines attempt to address the problem of coordinating with humans without collecting and using human data [Strouse *et al.*, 2021]. This is opposed to traditional approaches [Carroll *et al.*, 2019] where the adaptive agents should be trained with human proxies before cooperating with humans, which is often expensive. The automated process of generating a pool of artificial agents often benefits from approaches that encourage diversity, either by diversifying the population during training or by collecting agents. Towards enhancing the ability to generalise for artificial intelligence, this focuses on addressing the question of determining the data on which deep neural networks should undergo training to achieve effective generalisation [Yarats *et al.*, 2022; Jiang *et al.*, 2023], i.e. highlighting the role of training data to the adaptability of agents. In other words, if the model is exposed to diverse datasets, it can generalise and behave better in testing with situations that are unseen during training.

Maintaining diversity in the self-play pool. The first work proposed in this line is [Strouse *et al.*, 2021], in which the pool of agents is constructed by selecting agents at three different stages of learning. This method used self-play reinforcement learning [Tesauro, 1994; Silver *et al.*, 2018] to train independent self-play agents. After the training process, the self-play agents selected for the pool are diverse in their performance on the task. They have (a) low performance, i.e. at the beginning of training; (b) high performance, i.e. at the final stage of training; and (c) medium performance, i.e. have the rewards are the average of high and low performance. Although this is a simple approach, the adaptive agent trained on this pool surprisingly behaves well when coordinating with novel partners. Yu et al. [2022a] employed the human bias into the pool of agents via hand-crafting a reward system used in training, i.e. the desire or motivation of agents. For each self-play agent, the reward is computed by weighting pre-defined events. The weights are then randomised so that agents in the pool have different motivations during training. We note that the event-based reward system in this work is built based on human judgement of events in scenarios of Overcooked. Hence, this approach is specifically designed for the Overcooked benchmark and requires human effort to construct the diversity scores.

Maintaining diversity in the population-based training. Compared to self-play training, population-based training [Jaderberg *et al.*, 2017] allows more control over the population of collaborators to develop the pools. Direct measurement on the output of policies such as the Jensen-Shannon divergence is employed to create a strategy that obtains the population with agents that have diverse trajectories (TrajDiv) in [Lupu *et al.*, 2021]. In MEP [Zhao *et al.*, 2023], the entropy rewards were derived to encourage the agents to behave not only differently from others in the population but also differently from themselves. Utilising the intrinsic rewards for conditional policy to act differently according to different discrete inputs was common used in single reinforcement learning [Eysenbach *et al.*, 2018]. Szot et al. [2023] adopt this technique to create a pool of agents that has a specific identity and agents are trained to have different behaviours (BDP).

[1]We note that in case the actual entropy of agents' behaviour can be computed, the maximum behavioural predictability which is extensively studied in human mobility research [Song *et al.*, 2010; Lu *et al.*, 2013] can also be used as an alternative criterion within our framework.

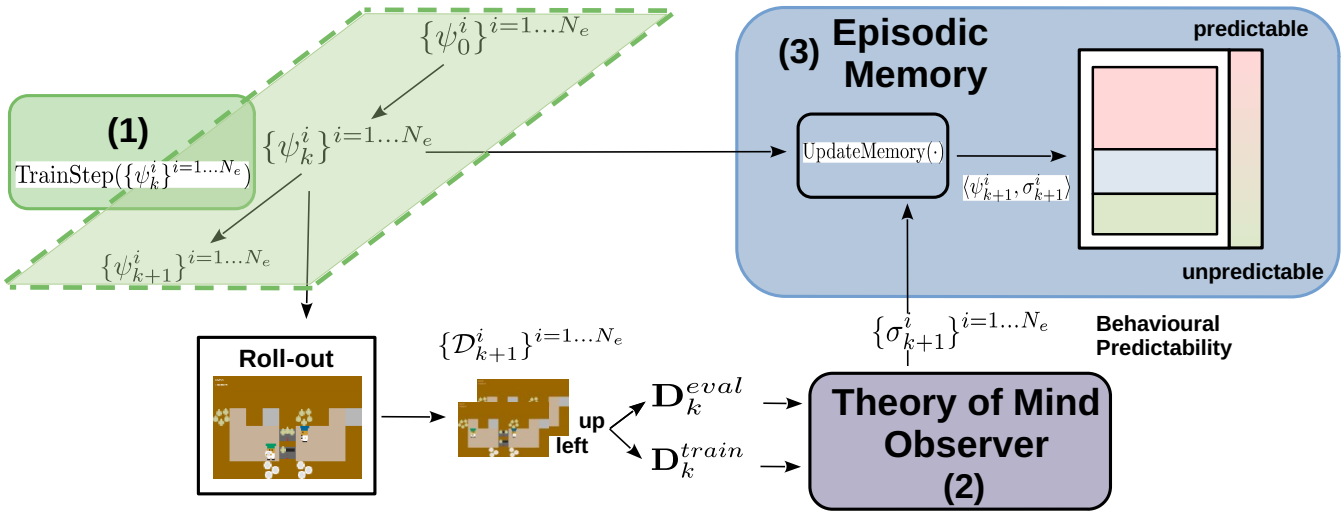

Figure 2: **Constructing the Training Pool with the support of the Theory of Mind Observer.** Our framework consists of three modules: (1) the generative training algorithm that is used to generate a population of collaborators; (2) the theory of mind observer that is continually trained on the behaviour data collected by rolling out collaborators in the population; and (3) the episodic memory which is constructed based on the level of behavioural predictability by storing the pairs of the weight and its LoBP score. The theory of mind observer computes the behavioural predictability so that the episodic memory can be updated with this information.

In [Lou *et al.*, 2023], the adaptive agent is trained frequently with versions of the policies in the pool during the training period. This method classified agents during the training process into different sets based on their performance to evenly select agents. All the listed methods do not explore the use of *behavioural predictability* as a criterion to select a diverse set of agents in the training pool.

Modelling other agents and machine theory of mind. Understanding others is a crucial ability for artificial agents to engage in social interactions [Albrecht and Stone, 2018]. This ability to attribute mental states, widely known as *theory of mind* in psychology research [Premack and Woodruff, 1978; Baron-Cohen *et al.*, 1985], has recently been studied in AI [Rabinowitz *et al.*, 2018]. Based on the assumption that novel partners seen during execution also are predicting others' behaviour, in [Knott *et al.*, 2021], the adaptive agent is trained to cooperate with a theory of mind-based policy that is cloned from human data. However, the machine theory of mind has not been used to maintain the diversity of behavioural predictability in the pool of training agents as in our work.

3 Preliminaries

3.1 Notations

Let us consider the multi-player Markov Decision Process (MDP) [Boutilier, 1996] as a tuple $\mathcal{M} = \langle \mathcal{N}, \mathcal{S}, \mathcal{A}, \mathcal{P}, \mathcal{R}, \gamma \rangle$ where $\mathcal{N} = \{1, \ldots, N\}$ is the set of N agents that operate in the scene, $\mathcal{S}$ is the state space, $\mathcal{A} = \mathcal{A}_1 \cup \cdots \cup \mathcal{A}_i \cdots \cup \mathcal{A}_N$ is the joint action space and $\mathcal{A}_i$ is the action space of the agent i, $\mathcal{P} : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \mapsto [0, 1]$ is the transition probability function, $\mathcal{R}$ is the reward function and γ is the discounted factor.

In this paper, we focus on tasks that involve two agents. For simplicity, we call them *the adaptive agent*, which is the agent that we can train to control during execution, and *the collaborator*, which is its partner that the adaptive agent needs to adapt to. It is worth noting that the adaptability of AI agents can be characterised as the ability to adapt to a variety of factors, such as environmental changes or objective changes. In our work, we consider the adaptive agent in social interactions, i.e. cooperating with different collaborators. At each time step t, each agent i will observe the state $s_t \in \mathcal{S}$ to take action $a_t^i \in \mathcal{A}_i$ accordingly. The joint action of agents is denoted as $\mathbf{a}_t \in \mathcal{A}$, and the trajectory of an agent at each time step t is $\tau_t = (s_0, \mathbf{a}_0, \ldots s_t, \mathbf{a}_t)$. Here, we study teaming in cooperative tasks; hence, the team will receive the team rewards $r_t \in \mathcal{R}$, with $\mathcal{R} : \mathcal{S} \times \mathcal{A} \mapsto \mathbb{R}$.

3.2 Zero-shot Coordination

The aim of a zero-shot coordination (ZSC) algorithm is to find a policy $\pi^{\text{adapt}} : \mathcal{S} \mapsto \mathcal{A}_i$ for the adaptive agent to cooperate well with unseen agents. Formally, let us consider a set $\mathcal{C}$ of all collaborators with their policies denoted as π^{colab}. This set consists of two sub-sets $\mathcal{C}_{\text{train}}$ which contains agents seen by the adaptive agent during training and $\mathcal{C}_{\text{hold-out}}$ which is the set of agents used during evaluation. When the adaptive agent is paired with the collaborator $\pi^{\text{colab}} \sim \mathcal{C}_{\text{hold-out}}$ sampled from the hold-out set, it acts to obtain the expected return of the team, hence, the objective of the adaptive agent π^{adapt} is

$$J(\pi^{\text{adapt}}, \pi^{\text{colab}}) = \mathbb{E}\left[\sum_{t=0}^{T} \gamma^t r_t(s_t, a_t^{\text{adapt}}, a_t^{\text{colab}})\right]$$

where T is the length of an episode. The adaptive policy that the zero-shot coordination algorithm seeks when trained with only collaborators sampled from $\mathcal{C}_{\text{train}}$ is

$$\pi_*^{\text{adapt}} = \arg\max_{\pi^{\text{adapt}}} \mathbb{E}_{\pi^{\text{colab}} \sim \mathcal{C}_{\text{hold-out}}}\left[J(\pi^{\text{adapt}}, \pi^{\text{colab}})\right].$$

3.3 Tackling Zero-shot Coordination via Generative Training Framework

To tackle the zero-shot coordination problem, recent works proposed using a generative strategy in training the adaptive agent [Strouse *et al.*, 2021; Lupu *et al.*, 2021; Szot *et al.*, 2023]. This training scheme has two stages as follows. In the *first stage*, a set (pool) of collaborators $\mathcal{C}_{\text{train}}$ are generated. Since this set of collaborators is only used in training the policy of the adaptive agent π^{adapt}, we denoted it as TrainingPool. In the *second stage*, the policy of the adaptive agent π^{adapt} is trained to cooperate with agents selected from this TrainingPool to achieve cooperative tasks. If the training pool is diverse, the policy of the adaptive agent π^{adapt} found in the second stage can cooperate with novel partners in evaluation.

We follow this two-stage training framework to train the adaptive agent. One distinct advantage of this framework is controlling the training data's diversity and quality. The generative procedure in the first stage allows us to manipulate or select the set of collaborators so that the adaptive agent can meet a diverse set of partners during training.

4 Proposed Approach

4.1 Overview

Our approach focuses on the *first stage*, i.e. generating the pool of training agents, of the two-stage training framework described in Section 3.3. Each agent in the pool is a policy which is parameterised by the weights, denoted by ψ. Given the set of N_e initial weights $\{\psi_0^i\}^{i=1...N_e}$, we aim to generate a new set of weights called $\text{TrainingPool} = \{\psi_h\}_{h=1...H}$, i.e. a pool of agents, with H is the size of the pool. This pool of H agents is then used to train an adaptive agent to cooperate with different types of novel partners.

We define the $\text{TrainStep}(\cdot)$ as a procedure that updates weights ψ_k of N_e agents, i.e. $\{\psi_{k+1}^i\}^{i=1...N_e} \leftarrow \text{TrainStep}(\{\psi_k^i\}^{i=1...N_e})$. Iteratively executing this training procedure will create different versions of agents. We denote the number of iterations as K. We note that the $\text{TrainStep}(\cdot)$ procedure can follow any tradition of updating the weights and go through multiple update steps. Here, we choose to use the RL self-play procedure [Silver *et al.*, 2018]. However, we do not restrict our approach to select agents induced by this type of policy training; therefore, one can explore using a population-based procedure under our framework.

We propose an algorithm that alternates between training and selecting the training weights to construct the TrainingPool. This is done by a theory of mind model and episodic memory. Briefly, at each iteration, we update N_e weights and store them in the memory via LoBP-based updating mechanism. So, our episodic memory keeps growing until we have H agents. After this, we replace agents to keep the memory size as H. We continually train a ToM model $f_\theta^{ToM}(\cdot)$ to predict the behaviour of policies on behavioural data collected from the population. Then, the behavioural predictability of each agent is computed based on comparing the prediction of the ToM model with the ground truth behaviour. Finally, we construct the episodic memory that uses this information to collect the weights to the training pool. The proposed episodic memory stores pairs of the weight and its LoBP score.

Algorithm 1: Constructing the Training Pool (Fig. 2)

Input : $\{\psi_0^i\}^{i=1...N_e}$ is the set of N_e randomly initialised weights; ToM Agent $f_\theta^{ToM}(\cdot)$; $\mathcal{M}$ is the episodic memory; K is the number of iterations.

Output: $\text{TrainingPool} = \{\psi_h\}_{h=1...H}$ with H is the number of collaborators in the training pool.

$\{\mathcal{D}_0^i\}^{i=1...N_e} \leftarrow \text{Roll-out}(\{\psi_0^i\}^{i=1...N_e})$;
$\mathbf{D}_0^{train}, \mathbf{D}_0^{eval} \leftarrow \{\mathcal{D}_0^i\}^{i=1...N_e}$;
$\{\sigma_0^i\}^{i=1...N_e} \leftarrow \text{ToMEval}(\mathbf{D}_0^{eval})$;
$\mathcal{M} \leftarrow \text{UpdateMemory}(\mathcal{M}, \{\langle\psi_0^i, \sigma_0^i\rangle\}^{i=1...N_e})$;
for $k \leftarrow 0$ **to** $K-1$ **do**
  Run the training procedure
  $\{\psi_{k+1}^i\}^{i=1...N_e} \leftarrow \text{TrainStep}(\{\psi_k^i\}^{i=1...N_e})$;
  // LoBP Assessment (4.2):
  $\{\mathcal{D}_{k+1}^i\}^{i=1...N_e} \leftarrow \text{Roll-out}(\{\psi_{k+1}^i\}^{i=1...N_e})$;
  $\mathbf{D}_{k+1}^{train}, \mathbf{D}_{k+1}^{eval} \leftarrow \{\mathcal{D}_{k+1}^i\}^{i=1...N_e}$;
  $\theta_{k+1} \leftarrow \text{ToMUpdate}(f_\theta^{\text{ToM}}(\cdot), \theta_k; \mathbf{D}_{k+1}^{train})$;
  $\{\sigma_{k+1}^i\}^{i=1...N_e} \leftarrow \text{ToMEval}(\mathbf{D}_{k+1}^{eval})$;
  // Update Episodic Memory (4.3):
  $\text{UpdateMemory}(\mathcal{M}, \{\langle\psi_{k+1}^i, \sigma_{k+1}^i\rangle\}^{i=1...N_e})$;
end
Collect all weights in $\mathcal{M}$ to the TrainingPool ;
return TrainingPool

The framework is shown in Fig. 1, and the overall algorithm is shown in Algo. 1. Details on the behavioural predictability and the ToM observer are presented in Section 4.2, and the episodic memory with its updating mechanism is introduced in Section 4.3.

4.2 Theory of Mind Observer and the Assessment of Behavioural Predictability

Behavioural Predictability

In this section, we will present how to compute the *behavioural predictability* of a policy, given information about the trajectories generated by this policy. This quantity is computed by *a theory of mind observer*, which is the model $f_\theta^{ToM}(\cdot)$ parameterised by the weight θ. The ToM observer is trained to predict the actions and intentions of others. Given a weight ψ_k^i which represents the version k of the agent i in the pool, we collect the *behavioural data* $\mathbf{D}_k^i$ by rolling out the policy with respect to the weight ψ_k^i in the environment to get M trajectories, then query the prediction about the next-step action and intention $(\tilde{a}_{t+1}, \tilde{g}_{t+1}) = f_\theta^{ToM}(\tau_t)$ from the theory of mind agent. The behavioural predictability of the policy with respect to the weight ψ_k^i is computed by

$$\sigma_k^i = -\frac{1}{M} \sum_{\tau_t \sim \mathbf{D}_k^i} \left[l_{\text{a}}(\tau_t) + l_{\text{g}}(\tau_t) \right], \tag{1}$$

where

$$l_{\text{a}}(\tau_t) = -\log p(\tilde{a}_{t+1} = a_{t+1} | \tau_t), \text{and}$$

$$l_{\mathrm{g}}(\tau_t) = -\log p(\tilde{g}_{t+1} = g_{t+1}|\tau_t)$$

are the cross-entropy losses of the ToM model in predicting next-step primitive actions l_{a} and in predicting goals or intentions l_{g}, respectively. Intuitively, if the trained observer does not predict well the behaviour generated by a policy (l_{a} or l_{g} is high), then this policy can be considered as having a low behaviour predictability (σ is low), and vice versa.

According to our criterion, two arbitrary agents x and y with weights ψ^x and ψ^y, respectively, are considered as *equal/similar in terms of LoBP* if the gap between their predictability scores σ^x and σ^y is sufficiently small, i.e. $|\sigma^x - \sigma^y| < \delta_{BP}$ for a small $\delta_{BP} \in \mathbb{R}^+$. We note that different approaches can be used to achieve diversity in behavioural predictability. One can build an agent model specifically for solving the task and use parameters to vary the predictability of agents in the pool. Due to simplicity, we propose to use episodic memory to promote LoBP diversity in Section 4.3.

Training the Theory of Mind Observer

In our approach, the ToM observer is continually trained along with the development of weights $\{\psi^i_{k+1}\}^{i=1\dots N_e}$. First, by rolling out policies with weights $\{\psi^i_{k+1}\}^{i=1\dots N_e}$ in the environment, we collected the *behavioural data* $\{\mathcal{D}^i_{k+1}\}^{i=1\dots N_e}$ as shown in the 8^{th} line of the Algo. 1. Second, this behavioural data obtained from the roll-out procedure $\{\mathcal{D}^i_{k+1}\}^{i=1\dots N_e}$ is divided into two sets $\mathbf{D}^{train}_{k+1}$ and $\mathbf{D}^{eval}_{k+1}$ by a fixed ratio (Alg. 1, line 9^{th}). Finally, the ToM observer is trained on the training set $\mathbf{D}^{train}_{k+1}$ ($\mathrm{ToMUpdate}(\cdot)$ in 10^{th} of Alg. 1). It then computes the behavioural predictability σ^i_k on the evaluation set $\mathbf{D}^{eval}_{k+1}$ ($\mathrm{ToMEval}(\cdot)$ in line 11^{th} of Alg. 1).

4.3 LoBP-based Episodic Memory

Design of LoBP-based Episodic Memory

We design the LoBP-based memory as an episodic memory $\mathcal{M}$ that contains H slots (equal the size of the $\mathrm{TrainingPool}$) and is organised into B segments. Each slot stores a pair of weights and the behavioural predictability score $\langle\psi, \sigma\rangle$.We refer to this pair as the *value* of the memory slot, which can be updated, i.e. written or erased. Each segment $S \in \{S_1, \dots, S_B\}$ contains several slots with a similar level of behavioural predictability as defined in Section 4.2. The number of segments B and the LoBP interval δ_{BP} are empirically chosen and fixed during the process.

Maintaining Diverse Level of Behaviour Predictability and Favouring Complex Behaviour

In this paper, we use a simple yet efficient mechanism to update the episodic memory and discard seen policies to guarantee a pool of agents with diversity in LoBP. The memory update mechanism, i.e. $\mathrm{UpdateMemory}(\cdot)$ procedure, is described in Algo. 2. In detail, when a new policy with its behavioural predictability score, i.e. $\langle\psi^i_k, \sigma^i_k\rangle$, comes to the update mechanism of the episodic memory, we will first erase the earliest memory value (making the slot empty) in the segment that has largest size (line 6^{th} of Algo. 2), then write the weight and its predictability score $\langle\psi^i_k, \sigma^i_k\rangle$ to the related segment. To guarantee there is no bias in the identity of the agent, we shuffle the agent set as in the 1^{st} line of Algo. 2.

Algorithm 2: $\mathrm{UpdateMemory}$

Input : $\{\psi^i_k\}^{i=1\dots N_e}$ is the set of N_e weights;
$\mathcal{M}$ is the episodic memory.
Output: $\mathcal{M}$

1 **for** i **in** $\mathrm{shuffle}(\{1 \dots N_e\})$ **do**
2 **if** $\mathcal{M}$ *is not full* **then**
3 $\mathcal{M} \leftarrow \langle\psi^i_k, \sigma^i_k\rangle$;
4 **else**
5 Discard the earliest came value $\langle\psi, \sigma\rangle_f$ to the segment that has the maximum size S_m, i.e. $\langle\psi, \sigma\rangle_f \in S_m$ s.t. $S_m = \arg\max_{S_j}(|S_j|)$ where $S_j \in \{S_1, \dots, S_B\}$ is the segment and $|S_j|$ is the size of S_j;
6 $\mathcal{M} \leftarrow \langle\psi^i_k, \sigma^i_k\rangle$;
7 **end**
8 **end**
9 **return** Updated $\mathcal{M}$

Diversity in the level of behavioural predictability. To exploit all the capacity of the memory, when the memory has not reached its limitation of size, we will write the pairs of the policy's weight and its behavioural predictability score $\langle\psi^i_k, \sigma^i_k\rangle$ into a slot in corresponding segments without any selection. When the episodic memory is full, to maintain the diversity in the LoBP and avoid overloading, we erase one *value* of the slot in the segment with the maximum size and write to the corresponding segment. We recall from the previous section that each segment in the episodic memory contains weights with similar levels of behavioural predictability.

Favouring complex and meaningful behaviour. Because behaviour displayed in the late stage of training is often more complex and meaningful than in the early stage, the update mechanism will keep later samples and discard earlier samples from memory. In other words, this mechanism results in first-in-first-out (FIFO) property for each segment.

5 Experiments

5.1 Experiment Setting

We conducted experiments on different scenarios of Overcooked which is a benchmark to assess the zero-shot coordination ability of the adaptive agents. In this game, there are two agents that need to collaborate to achieve a cooking task. The Overcooked game was first introduced to study the coordination of AI in [Carroll *et al.*, 2019]. Each agent can take one action in the set of 6 actions $\mathcal{A} = \{\mathrm{up}, \mathrm{down}, \mathrm{left}, \mathrm{right}, \mathrm{interact}\}$. During one episode, agents try to deliver as many dishes as possible before the end of the episode, reaching an end after $T = 400$ timesteps. To make one dish, agents have to collect ingredients and put them into the pot to cook. After 20 time steps of cooking on the stove, the dish is ready and can be delivered to a counter located at a fixed place in the room. In our experiments, all scenarios (Asymmetric Advantages, Coordination Ring) have sparse rewards, i.e. the team reward is only given to agents at the delivery time (successfully achieving the task).

	FCP	TrajDiv	MEP	BDP	Our	Our (intent.)
Asym.	271.33 (23.88)	251.714 (17.19)	273.636 (19.92)	278.484 (14.29)	**293.55 (10.77)**	**298.824 (9.86)**
Coord.	103.444 (10.02)	107.54 (9.075)	111.9 (8.076)	112.3 (16.537)	**120.87 (6.84)**	**120.37 (4.34)**

Table 1: Rewards in tasks when adaptive agents are paired with reinforcement learning agents trained via self-play process.

	FCP	TrajDiv	MEP	BDP	Our	Our (intention)
Onion and Delivery	310.67	286.0	333.33	321.33	333.33	328.66
Delivery	307.33	346.0	216.67	318.00	346.0	322.0
Onion Everywhere	215.33	253.33	230.0	255.00	248.0	270.67
avg.	277.78	295.11	260.00	298.0	**309.11**	**307.11**

Table 2: Rewards in tasks when adaptive agents are paired with scripted agents with different behaviour preferences.

5.2 Baselines

We consider state-of-the-art baselines in two branches of approaches that employ the generative two-stage training framework. They are different in classes of training methods used in stage 1: (1) the self-play (SP) training and (2) the population-based training (PBT).

- SP training method: fictitious co-play (FCP) [Strouse *et al.*, 2021] that used RL SP strategy to train and a reward-based rule to select collaborators into the training pool.
- Population-based training method: (1) Trajectory Diversity (TrajDiv) [Lupu *et al.*, 2021]; (2) Maximum entropy population-based training (MEP) [Zhao *et al.*, 2023]; and (3) Behaviour diversity play (BDP) [Szot *et al.*, 2023]. All three methods utilise PBT to develop the training pool but use different criteria for maintaining the diversity of the pool.

We consider two versions of our approaches where the LoBP score is only computed by action prediction, i.e. there is no l_g in Eq. 1, and where both action and intention prediction are taken into account. To evaluate our approaches, we followed the literature on experimental designs. We first compare adaptive agents in cooperating with reinforcement learning agents that are developing via self-play procedure. We used different seeds to generate the pool of agents to train the agents used to evaluate the adaptive agents in the test phase. We then evaluate the coordination skill of adaptive agents with the scripted agents and the human proxies.

5.3 Cooperating with RL Agents

To evaluate the ability to cooperate with novel agents, we trained a new set of RL agents by self-play training. This set is trained with different seeds from the training process of SP agents in the FCP pool and our pool of agents. The new RL self-play agents used to evaluate the adaptive agents are randomly selected during SP training. Hence, although they shared the same training algorithm (RL self-play process), the sets of agents used in train ($\mathcal{C}_{\text{train}}$) and test ($\mathcal{C}_{\text{hold-out}}$) are different. Our approach outperforms other baselines on this task. During SP training, the RL agents exhibit behaviours with diverse predictability while learning skills. Therefore, training by our approach helps social agents to adapt well to these behaviours. The performances when pairing adaptive agents with hold-out RL agents are shown in Table 1.

5.4 Cooperating with Scripted Agents

In this section, we constructed scripted agents that have high behaviour preferences. We considered interactions between the adaptive agent and three types of behaviour: (1) placing onion in pot and delivery when the soup is cooked; (2) delivery when the soup is cooked; and (3) putting onion everywhere. While the first two types of agents have meaningful behaviour, the third procedural agent's behaviour obstructs the team from achieving high performance. This type of *useless predictable* behaviour can appear in the RL self-play training, however, they often in the early phase and will be discarded from the episodic memory due to the *usefull predictable* behaviour when the SP agents begin to learn near-optimal behaviour. More visualisation of this phenomenon is shown in Fig. 4. The results are shown in Table 2.

5.5 Cooperating with Human Proxy Agents

General Human Proxy

We use the behaviour cloning method [Bain and Sammut, 1995] to train the policy π^{proxy} that imitates the behaviour of humans in scenarios of Overcooked. These policies are referred to the *human proxies*. The human data used in our works are human interactions in experiments established in [Carroll *et al.*, 2019]. The results are shown in Table 3 (performance of our method is higher and statistically significant different to performance of baselines with $p < 0.05$).

Specific Human Proxies

We also evaluate the adaptive agents on different human proxies. To construct different human proxies, we first meta-trained a general proxy $\bar{\pi}_{proxy}$ on trajectories of 12 individuals. We then fine-tuned this $\bar{\pi}^{proxy}$ on trajectories of others 6 individuals to obtain the specific proxies for each of them $\{\pi_0^{proxy}, \pi_1^{proxy}, \pi_2^{proxy}, \pi_3^{proxy}, \pi_4^{proxy}, \pi_5^{proxy}\}$. This type of training is adopted from [Duan *et al.*, 2017]. These policies are assumed to reflect the behaviour of individuals. The results are shown in Figure 3. Our approaches produce adaptive agents that gain higher team rewards on average when cooperating with these policies.

	FCP	TrajDiv	MEP	BDP	Our	Our (intent.)
Asym.	229.33 (24.02)	231.862 (25.26)	240.73 (23.66)	254.6 (17.99)	260.864 (14.54)	**276.796 (14.31)**
Coord.	88.466 (10.36)	100.266 (11.69)	103.934 (2.14)	105.266 (13.33)	102.8 (14.73)	**114.4 (15.00)**

Table 3: Rewards obtained in tasks when adaptive agents are paired with human proxy.

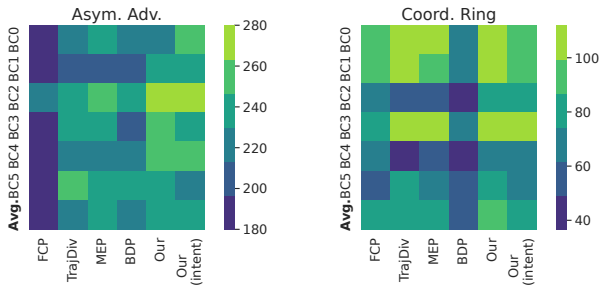

Figure 3: Rewards obtained when the adaptive agent that is trained by our approach and other baselines is paired with different specific human proxies $\{\pi_j^{proxy}\}^{j=0...5}$. The lighter colour means the higher team reward obtained.

	Low Perf.	High Perf.
FCP	12	12 (med) + 12 (high)
Our	7	29
Our (intent)	5	31

Table 4: Comparison of portions in the pool of agents. Our method selects a small number of random agents than FCP.

5.6 Analysis of the Pool of Agents

Agents selected by Level of Behavioural Predictability

In this section, we analyse in detail the pool of agents that are generated by using the LoBP criteria. Fig. 4(a) shows that all SP agents were chosen to the training pool. Our training pool contains agents that are diverse in the level of behavioural predictability (Fig. 4(b)). To analyse the difference between the FCP pool and the LoPB-based pool, we divided the training process of the SP agents into three stages. The early stage is the period before the SP agent obtains half of the maximum training rewards. Following this stage are the middle and high stages where the SP agents obtain higher rewards. Table 4 shows that our pool mainly contains the weights of agents in the middle and high stages of training, i.e. the number of high-performance and meaningful in our pool is higher than the FCP pool. This helps the adaptive agents focus more on learning to cooperate with the agents that deliver complex behaviour instead of random behaviour often appearing in the early training phase. To further discussion, training good ToM observer can improve the efficiency of our approach, i.e. the ability of the ToM observer also affects to the settings where there are more agents or complex environment.

Pure Pools Comparison

We further compared our pool with two other pools: Init Pool, in which all agents act unpredictably, and Final Pool, in which all agents are well-trained. The results are shown in Fig. 5. We evaluate adaptive agents on three sets: (1) a set of random

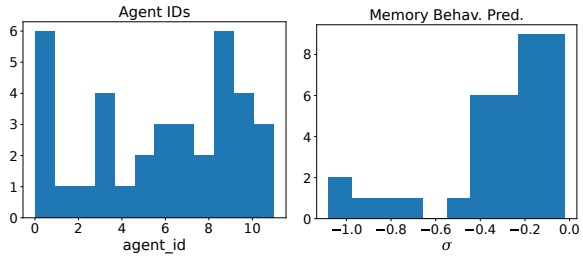

Figure 4: Episodic memory contents (y-axis indicates the number of agents). All agents are chosen (a) and the pool contains behaviours with different level of predictability σ (as shown in (b)).

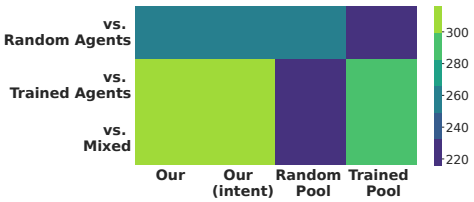

Figure 5: Comparison of rewards obtained when the adaptive agents that were trained with our training pool (first two columns), with a pool that contains only unpredictable agents (3^{rd} column) and a pool that have all well-trained agents (4^{th} column) are paired with agents that are unpredictable (1^{st} row), well-trained (2^{nd} row), and mixed behaviour (3^{rd} row). The lighter colour the higher reward.

agents (Fig. 5 (1^{st} row)), a set of trained agents (Fig. 5 (2^{nd} row)), and a mix of different reinforcement learning agents during self-play training (Fig. 5 (3^{rd} row)). There is a contrast between the performance of adaptive agents trained on a random pool and the trained pool because they focus on different types of collaborators. The adaptive agent trained with only pool that have unpredictable behaviour behaves worst. Our adaptive agents obtained better results when paired with trained and mixed agents, and better performance than the adaptive agent trained on a well-trained pool when paired with random agents.

6 Conclusions

In this paper, we tackle the ZSC by the generative training framework, in which a pool of agents is auto-generated in the first stage and the adaptive agent is trained to cooperate with agents in this pool to achieve the task in the second stage. We focus on the first stage and propose to use a novel way to promote the diversity of the training pool using the levels of behavioural predictability. To realise the concept, we design a ToM observer equipped with a episodic memory to monitor the pool generation process. We demonstrated that agents trained to cooperate with this training pool can better adapt to novel partner meaning they are more robust and trustworthy compared to diversity based methods.

Acknowledgements

This research was partially funded by the Australian Government through the Australian Research Council's Discovery Project funding scheme (project DP210102798). The views expressed herein are those of authors and are not necessarily those of the Australian Government or Australian Research Council.

References

[Albrecht and Stone, 2018] Stefano V Albrecht and Peter Stone. Autonomous agents modelling other agents: A comprehensive survey and open problems. *Artificial Intelligence*, 258:66–95, 2018.

[Bain and Sammut, 1995] Michael Bain and Claude Sammut. A framework for behavioural cloning. In *Machine Intelligence 15*, pages 103–129, 1995.

[Bard *et al.*, 2020] Nolan Bard, Jakob N Foerster, Sarath Chandar, Neil Burch, Marc Lanctot, H Francis Song, Emilio Parisotto, Vincent Dumoulin, Subhodeep Moitra, Edward Hughes, et al. The hanabi challenge: A new frontier for ai research. *Artificial Intelligence*, 280:103216, 2020.

[Baron-Cohen *et al.*, 1985] Simon Baron-Cohen, Alan M Leslie, and Uta Frith. Does the autistic child have a "theory of mind"? *Cognition*, 21(1):37–46, 1985.

[Boutilier, 1996] Craig Boutilier. Planning, learning and coordination in multiagent decision processes. In *TARK*, volume 96, pages 195–210. Citeseer, 1996.

[Bowling and McCracken, 2005] Michael Bowling and Peter McCracken. Coordination and adaptation in impromptu teams. In *AAAI*, volume 5, pages 53–58, 2005.

[Brown *et al.*, 2020] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.

[Carroll *et al.*, 2019] Micah Carroll, Rohin Shah, Mark K Ho, Tom Griffiths, Sanjit Seshia, Pieter Abbeel, and Anca Dragan. On the utility of learning about humans for human-ai coordination. *Advances in Neural Information Processing Systems*, 32, 2019.

[Dafoe *et al.*, 2021] Allan Dafoe, Yoram Bachrach, Gillian Hadfield, Eric Horvitz, Kate Larson, and Thore Graepel. Cooperative ai: machines must learn to find common ground. *Nature*, 593(7857):33–36, 2021.

[Duan *et al.*, 2017] Yan Duan, Marcin Andrychowicz, Bradly Stadie, OpenAI Jonathan Ho, Jonas Schneider, Ilya Sutskever, Pieter Abbeel, and Wojciech Zaremba. One-shot imitation learning. *Advances in neural information processing systems*, 30, 2017.

[Eysenbach *et al.*, 2018] Benjamin Eysenbach, Abhishek Gupta, Julian Ibarz, and Sergey Levine. Diversity is all you need: Learning skills without a reward function. In *International Conference on Learning Representations*, 2018.

[Jaderberg *et al.*, 2017] Max Jaderberg, Valentin Dalibard, Simon Osindero, Wojciech M Czarnecki, Jeff Donahue, Ali Razavi, Oriol Vinyals, Tim Green, Iain Dunning, Karen Simonyan, et al. Population based training of neural networks. *arXiv preprint arXiv:1711.09846*, 2017.

[Jakubik *et al.*, 2022] Johannes Jakubik, Michael Vössing, Niklas Kühl, Jannis Walk, and Gerhard Satzger. Data-centric artificial intelligence. *arXiv preprint arXiv:2212.11854*, 2022.

[Jiang *et al.*, 2023] Minqi Jiang, Tim Rocktäschel, and Edward Grefenstette. General intelligence requires rethinking exploration. *Royal Society Open Science*, 10(6):230539, 2023.

[Knott *et al.*, 2021] Paul Knott, Micah Carroll, Sam Devlin, Kamil Ciosek, Katja Hofmann, Anca Dragan, and Rohin Shah. Evaluating the robustness of collaborative agents. In *Proceedings of the 20th International Conference on Autonomous Agents and MultiAgent Systems*, pages 1560–1562, 2021.

[Lou *et al.*, 2023] Xingzhou Lou, Jiaxian Guo, Junge Zhang, Jun Wang, Kaiqi Huang, and Yali Du. PECAN: Leveraging policy ensemble for context-aware zero-shot human-ai coordination. In *Proceedings of the 2023 International Conference on Autonomous Agents and Multiagent Systems*, pages 679–688, 2023.

[Lu *et al.*, 2013] Xin Lu, Erik Wetter, Nita Bharti, Andrew J Tatem, and Linus Bengtsson. Approaching the limit of predictability in human mobility. *Scientific reports*, 3(1):2923, 2013.

[Lupu *et al.*, 2021] Andrei Lupu, Brandon Cui, Hengyuan Hu, and Jakob Foerster. Trajectory diversity for zero-shot coordination. In *International conference on machine learning*, pages 7204–7213. PMLR, 2021.

[Mazumder *et al.*, 2022] Mark Mazumder, Colby Banbury, Xiaozhe Yao, Bojan Karlaš, William Gaviria Rojas, Sudnya Diamos, Greg Diamos, Lynn He, Alicia Parrish, Hannah Rose Kirk, et al. Dataperf: Benchmarks for data-centric ai development. *arXiv preprint arXiv:2207.10062*, 2022.

[Mirsky *et al.*, 2022] Reuth Mirsky, Ignacio Carlucho, Arrasy Rahman, Elliot Fosong, William Macke, Mohan Sridharan, Peter Stone, and Stefano V Albrecht. A survey of ad hoc teamwork: Definitions, methods, and open problems. In *European Conference on Multiagent Systems*, 2022.

[Papoudakis *et al.*, 2021] Georgios Papoudakis, Filippos Christianos, and Stefano Albrecht. Agent modelling under partial observability for deep reinforcement learning. *Advances in Neural Information Processing Systems*, 34:19210–19222, 2021.

[Paul *et al.*, 2021] Mansheej Paul, Surya Ganguli, and Gintare Karolina Dziugaite. Deep learning on a data diet: Finding important examples early in training. *Advances in Neural Information Processing Systems*, 34:20596–20607, 2021.

[Premack and Woodruff, 1978] David Premack and Guy Woodruff. Does the chimpanzee have a theory of mind? *Behavioral and brain sciences*, 1(4):515–526, 1978.

[Rabinowitz *et al.*, 2018] Neil Rabinowitz, Frank Perbet, Francis Song, Chiyuan Zhang, SM Ali Eslami, and Matthew Botvinick. Machine theory of mind. In *International conference on machine learning*, pages 4218–4227. PMLR, 2018.

[Silver *et al.*, 2018] David Silver, Thomas Hubert, Julian Schrittwieser, Ioannis Antonoglou, Matthew Lai, Arthur Guez, Marc Lanctot, Laurent Sifre, Dharshan Kumaran, Thore Graepel, et al. A general reinforcement learning algorithm that masters chess, shogi, and go through self-play. *Science*, 362(6419):1140–1144, 2018.

[Song *et al.*, 2010] Chaoming Song, Zehui Qu, Nicholas Blumm, and Albert-László Barabási. Limits of predictability in human mobility. *Science*, 327(5968):1018–1021, 2010.

[Stone *et al.*, 2010] Peter Stone, Gal Kaminka, Sarit Kraus, and Jeffrey Rosenschein. Ad hoc autonomous agent teams: Collaboration without pre-coordination. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 24, pages 1504–1509, 2010.

[Strouse *et al.*, 2021] DJ Strouse, Kevin McKee, Matt Botvinick, Edward Hughes, and Richard Everett. Collaborating with humans without human data. *Advances in Neural Information Processing Systems*, 34:14502–14515, 2021.

[Szot *et al.*, 2023] Andrew Szot, Unnat Jain, Dhruv Batra, Zsolt Kira, Ruta Desai, and Akshara Rai. Adaptive coordination in social embodied rearrangement. In Andreas Krause, Emma Brunskill, Kyunghyun Cho, Barbara Engelhardt, Sivan Sabato, and Jonathan Scarlett, editors, *Proceedings of the 40th International Conference on Machine Learning*, volume 202, pages 33365–33380. PMLR, 23–29 Jul 2023.

[Tesauro, 1994] Gerald Tesauro. Td-gammon, a self-teaching backgammon program, achieves master-level play. *Neural computation*, 6(2):215–219, 1994.

[Treutlein *et al.*, 2021] Johannes Treutlein, Michael Dennis, Caspar Oesterheld, and Jakob Foerster. A new formalism, method and open issues for zero-shot coordination. In *International Conference on Machine Learning*, pages 10413–10423. PMLR, 2021.

[Whang *et al.*, 2023] Steven Euijong Whang, Yuji Roh, Hwanjun Song, and Jae-Gil Lee. Data collection and quality challenges in deep learning: A data-centric ai perspective. *The VLDB Journal*, 32(4):791–813, 2023.

[Yarats *et al.*, 2022] Denis Yarats, David Brandfonbrener, Hao Liu, Michael Laskin, Pieter Abbeel, Alessandro Lazaric, and Lerrel Pinto. Don't change the algorithm, change the data: Exploratory data for offline reinforcement learning. *arXiv preprint arXiv:2201.13425*, 2022.

[Yu *et al.*, 2022a] Chao Yu, Jiaxuan Gao, Weilin Liu, Botian Xu, Hao Tang, Jiaqi Yang, Yu Wang, and Yi Wu. Learning zero-shot cooperation with humans, assuming humans are biased. In *The Eleventh International Conference on Learning Representations*, 2022.

[Yu *et al.*, 2022b] Yu Yu, Shahram Khadivi, and Jia Xu. Can data diversity enhance learning generalization? In *Proceedings of the 29th international conference on computational linguistics*, pages 4933–4945, 2022.

[Zha *et al.*, 2023a] Daochen Zha, Zaid Pervaiz Bhat, Kwei-Herng Lai, Fan Yang, and Xia Hu. Data-centric ai: Perspectives and challenges. In *Proceedings of the 2023 SIAM International Conference on Data Mining (SDM)*, pages 945–948. SIAM, 2023.

[Zha *et al.*, 2023b] Daochen Zha, Zaid Pervaiz Bhat, Kwei-Herng Lai, Fan Yang, Zhimeng Jiang, Shaochen Zhong, and Xia Hu. Data-centric artificial intelligence: A survey. *arXiv preprint arXiv:2303.10158*, 2023.

[Zhao *et al.*, 2023] Rui Zhao, Jinming Song, Yufeng Yuan, Haifeng Hu, Yang Gao, Yi Wu, Zhongqian Sun, and Wei Yang. Maximum entropy population-based training for zero-shot human-ai coordination. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 6145–6153, 2023.