Federated Agent Reinforcement Learning

Canyu Chen [*1] **Kangyu Zhu** [*2] **Zhaorun Chen** [3] **Zhanhui Zhou** [4] **Shizhe Diao** [5] **Yiping Lu** [1] **Tian Li** [3]
Manling Li [+1] **Dawn Song** [+4]

Abstract

Autonomous AI Agents powered by LLMs have shown remarkable abilities in diverse domains. However, the training process typically require *centralized* collection of large amounts of real-world user data, posing substantial privacy and regulatory concerns. To this end, we explore a new *decentralized* training paradigm, namely **FEDAGENT** (Federated Agent Reinforcement Learning), which enables collaborative learning of AI agents across distributed clients without sharing local data. Moreover, we construct the first decentralized agent learning environment **FEDAGENTGYM**, which includes four types of LLM agents, two application scenarios (WebShop and ALFWorld), three variations of decentralized settings, and three newly defined heterogeneity challenges (*Preference Heterogeneity*, *Coverage Heterogeneity*, and *Hardness Heterogeneity*), to systematically investigate its effectiveness and impact factors. Extensive empirical studies show that FEDAGENT can have comparable performance to centralized agent training and exhibit strong robustness against heterogeneities, which shows the feasibility of training AI agents while protecting data privacy and opens a new direction for agent learning. The code is available here.

1. Introduction

The rapid advancement of AI agents, especially those powered by Large Language Models (LLMs), has demonstrated remarkable capabilities across diverse domains, from web navigation to embodied environments (Zhang et al., 2026a; Gao et al., 2025; Liu et al., 2025a). However, training these agents typically requires *centralized* access to vast amounts of users' real-world task query and trajectory data, which are inherently privacy-sensitive and hard to acquire due to regulatory compliance. Thus, a foundational question is: ***Can we train AI agents while protecting users' data privacy?***

In this paper, we explore a new *decentralized* training paradigm, namely **FEDAGENT** (**Federated Agent Reinforcement Learning**), which enables collaborative learning of AI agents, particularly LLMs, across distributed clients without sharing local data. In each round, the server distributes the current model to selected clients, who then train locally on their own data and send back their updated models. The server aggregates these updates by averaging them to create an improved global model for the next round. This process repeats iteratively, facilitating distributed LLM agent training while preserving data privacy since only model parameters, not raw data, are exchanged.

Compared with the previous federated learning literature, FEDAGENT is faced with fundamentally new challenges. The majority of existing federated learning research has concentrated on supervised classification tasks. There are also recent works that have explored federated reinforcement learning (FRL) for traditional RL settings (Qi et al., 2021; Kairouz et al., 2021; Liu et al., 2024a). However, both of them operate under distinct assumptions compared to LLM agent learning. Supervised federated learning is usually built on static data distributions and one-shot predictions, while traditional FRL typically assumes simple rewards, low-dimensional state and action spaces. In contrast, LLM agent learning involves *natural language state and action spaces, diverse task formulations, and complex environment interactions*, which create entirely new challenges for federated paradigms. Thus, another essential research question is: ***Is FedAgent really effective for training LLM agents?***

To systematically investigate the effectiveness of this new training paradigm as well as the impact factors, we built the first decentralized agent learning environment **FEDAGENTGYM**, which incorporates four types of LLM agents from different model families (Qwen and Llama) and with different scales (1.5B, 3B, and 7B), two application domains (WebShop and ALFWorld), three dimensions of variation in decentralized settings (the number of samples per client, the number of clients selected per communication round, and the number of local training epochs per client per round).

[*]Equal contribution. [+]Equal advising. [1]Northwestern University [2]Brown University [3]The University of Chicago [4]University of California, Berkeley [5]NVIDIA Research. Correspondence to: Canyu Chen <canyuchen@u.northwestern.edu>, Manling Li <manling.li@northwestern.edu>, Dawn Song <dawnsong@berkeley.edu>.

Preprint.

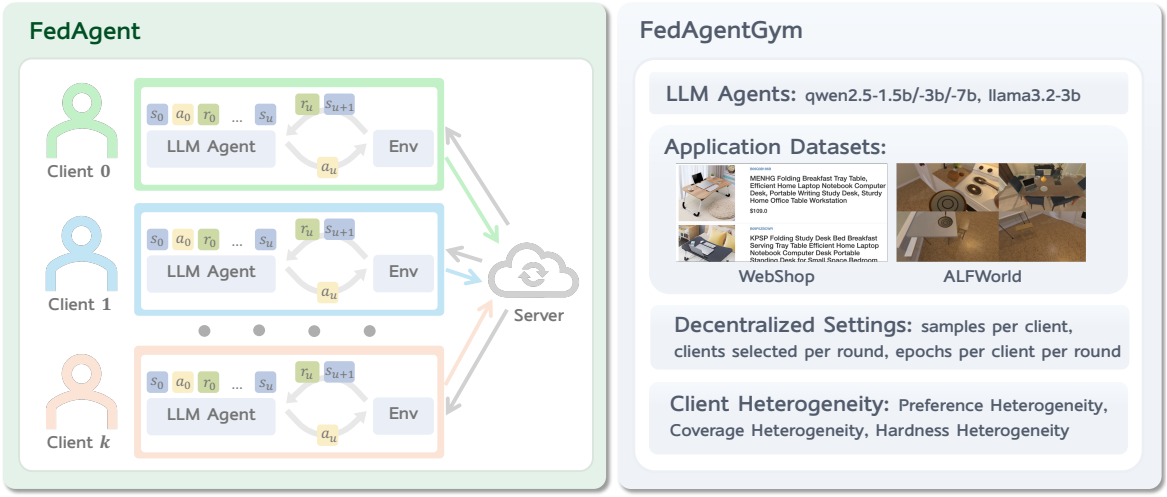


Figure 1: An Illustration of FEDAGENT and FEDAGENTGYM.

Importantly, since the existing heterogeneity challenges in federated learning have mostly been defined in the context of supervised classification tasks (Gao et al., 2022; Ye et al., 2024a), which focus on label skew, feature shift, or quantity imbalance, they are not directly applicable to LLM agent learning. Thus, we propose three new and orthogonal agent-specific heterogeneity definitions: ***Preference Heterogeneity***, where clients may prefer distinct types of tasks; ***Coverage Heterogeneity***, where the task sampling scope may vary across clients; ***Hardness Heterogeneity***, where the overall difficulty of tasks may differ among clients. Moreover, we carefully design three novel client partitioning strategies PREFERENCEPARTITION, COVERAGEPARTITION, and HARDNESSPARTITION accordingly, grounded in mathematical techniques such as *Gaussian Noise*, *Multinomial Sampling* and *Beta Distribution*. These strategies allow us to precisely control the extent of one type of heterogeneity across clients with a single hyperparameter, while keeping the other characteristics of the client distribution unchanged. We then incorporate them into FEDAGENTGYM to isolate and analyze the impact of each form of heterogeneity on FEDAGENT separately and controllably.

To investigate the effectiveness of FEDAGENT, we conduct an extensive and systematic study with FEDAGENTGYM. By training multiple LLM agents in the two application domains, we demonstrate that FEDAGENT consistently outperforms *local agent training* and can match the performance of *centralized agent training*, despite never sharing local data. By training LLM agents under different decentralized configurations, we discover the sensitivity patterns of FEDAGENT. Through our designed client partition strategies, we show FEDAGENT exhibits strong robustness to the aforementioned preference, coverage, and hardness heterogeneity challenges. Overall, our studies show the potential of scalable agent learning without sacrificing data privacy, provide valuable insights that inform practical deployment, and open new research directions in the field of agent learning.

Our contributions can be summarized as follows:

- We made the first attempt to formulate and explore a new decentralized paradigm of training AI agents, namely **FEDAGENT** (**Federated Agent Reinforcement Learning**), which enables collaborative agent learning across distributed clients without sharing local data.
- We constructed the first decentralized agent learning environment **FEDAGENTGYM**, which includes four types of LLM agents, two applications (WebShop and ALFWorld), three variations of decentralized settings, and three heterogeneity challenges, to analyze the performance of FEDAGENT systematically and controllably, and offer insights to guide future development.
- We propose to categorize the new client heterogeneity challenges in decentralized agent learning into *Preference Heterogeneity*, *Coverage Heterogeneity*, and *Hardness Heterogeneity*. To investigate how each type of heterogeneity affects the performance, we introduce three novel client partitioning methods: PREFERENCEPARTITION, COVERAGEPARTITION, and HARDNESSPARTITION.
- Through extensive studies on FEDAGENTGYM, we provide three key insights on the effectiveness of FEDAGENT: (1) FEDAGENT not only outperforms single-client local training but also achieves comparable performance to centralized agent learning. (2) FEDAGENT's effectiveness depends on decentralized configurations such as the number of clients selected per round and the number of epochs per client per round. (3) FEDAGENT shows high robustness against the agent-specific heterogeneity challenges.
- We release our code and environment as an extendable open-source library to inspire more future works in this new direction. The link to the repository is available here.

2. FEDAGENT: Federated Agent Reinforcement Learning

As shown in Algorithm 1, we consider a federated reinforcement learning setup for FEDAGENT. A population of clients are indexed by $k \in [K] = \{0, \dots, K-1\}$. Training proceeds for communication rounds $t = 0, \dots, T-1$. At round t, the server samples a subset $S_t \subset [K]$ of size $|S_t| = M$ uniformly without replacement, broadcasts the current global policy parameters θ_t, and aggregates the participating clients' locally updated parameters.

LLM Agent Training. The agent is a parametric policy π_θ (an LLM) that, conditioned on a task description c and an interaction history h_u up to step u, produces an action $a_u \sim \pi_\theta(\cdot \mid h_u, c)$. An action often contains both *a sequence of free-form tokens* (*i.e.*, the agent's intermediate reasoning) and *environment-facing choice* (*e.g.*, tool API calling). Each client k operates in a Markov Decision Process (MDP) environment $\mathcal{M}_k = (\mathcal{S}_k, \mathcal{A}_k, P_k, r_k, \rho_k, \gamma)$ with state space $\mathcal{S}_k$, action space $\mathcal{A}_k$, transition kernel P_k, reward function r_k, initial-state distribution ρ_k, and discount $\gamma \in (0, 1]$. Client k also has a distribution $\mathcal{D}_k$ over textual task descriptions $c \in \mathcal{C}_k$. Fix k and a task description $c \sim \mathcal{D}_k$. The agent interacts with $\mathcal{M}_k$ for horizon H, producing a trajectory:

$$\chi = (c, s_0, a_0, r_0, \dots, s_H), s_0 \sim \rho_k(\cdot \mid c)$$

$$a_u \sim \pi_\theta(\cdot \mid h_u, c),\ s_{u+1} \sim P_k(\cdot \mid s_u, a_u, c).$$

The discounted return of χ is $R(\chi) = \sum_{u=0}^{H-1} \gamma^u r_u$. It is worth noting that when the LLM agents generate H consecutive textual actions $(a_0, ..., a_{(H-1)})$ in a trajectory χ, each action may span thousands of tokens, considering LLM agents' long reasoning capacity (DeepSeek-AI et al., 2025). This makes token-level credit assignment across the trajectory particularly challenging.

Local objective (client k). Client k aims to maximize the expected episodic return of the policy on its own environments and tasks:

$$J_k(\theta) = \mathbb{E}_{c \sim \mathcal{D}_k}\, \mathbb{E}_{\chi \sim (\pi_\theta, \mathcal{M}_k, c)} \big[R(\chi)\big]. \tag{1}$$

During round t, each participating client initializes a local iterate at the broadcast model, $\theta_{k,t,0} \leftarrow \theta_t$, and performs τ steps of stochastic policy optimization. At local step $i \in \{0, \dots, \tau-1\}$, the client collects a batch of trajectories $B_{k,t,i} = \{\chi^{(b)}\}_{b=1}^{N_{k,t,i}}$ by interacting with $\mathcal{M}_k$ under $\pi_{\theta_{k,t,i}}$ and computes a policy-gradient estimate:

$$g_{k,t,i} = \nabla_\theta \widehat{J}_k(\theta_{k,t,i}; B_{k,t,i}) = \frac{1}{N_{k,t,i}} \sum_{b=1}^{N_{k,t,i}} \sum_{u=0}^{H(\chi^{(b)})-1} \nabla_\theta \log \pi_{\theta_{k,t,i}}\big(a_u^{(b)} \mid h_u^{(b)}, c^{(b)}\big)\, \widehat{A}_u^{(b)}. \tag{2}$$

Algorithm 1 FEDAGENT with Client and Server training

Require: Total clients K, rounds T, clients-per-round M, local steps τ, learning rate η
Ensure: Final LLM-based global policy parameters θ_{final}

Initialize global policy parameters θ_0 (an LLM)
for $t = 0$ **to** $T-1$ **do**
  Server: sample client subset $S_t \subset [K]$ with $|S_t| = M$ (uniform without replacement)
  Server: broadcast θ_t to all $k \in S_t$
  for each $k \in S_t$ **in parallel do**
    Set local iterate $\theta_{k,t,0} \leftarrow \theta_t$
    for $i = 0$ **to** $\tau - 1$ **do**
      Collect a mini batch of trajectories $B_{k,t,i}$ using policy $\pi_{\theta_{k,t,i}}$ in environment $\mathcal{M}_k$
      Estimate policy gradient for $J_k(\theta_{k,t,i})$ on client k:

$$g_{k,t,i} \leftarrow \nabla_\theta \hat{J}_k(\theta_{k,t,i}; B_{k,t,i}) \quad (\textit{e.g.}, \text{GRPO})$$

      Local update: $\theta_{k,t,i+1} \leftarrow \theta_{k,t,i} + \eta\, g_{k,t,i}$
    end for
    Client returns local model $\theta_{k,t,\tau}$ ▷ equivalently $\Delta\theta_{k,t} = \theta_{k,t,\tau} - \theta_t$
  end for
  Server: Aggregation via model averaging:

$$\theta_{t+1} \leftarrow \frac{1}{M} \sum_{k \in S_t} \theta_{k,t,\tau}$$

  (equivalently $\theta_{t+1} = \theta_t + \frac{1}{M} \sum_{k \in S_t} (\theta_{k,t,\tau} - \theta_t)$)
end for
return $\theta_{\text{final}} \leftarrow \theta_T$

where $\widehat{A}_u^{(b)}$ is any valid return/advantage signal (*e.g.*, a GRPO-style estimator (Shao et al., 2024)). The local update is $\theta_{k,t,i+1} = \theta_{k,t,i} + \eta\, g_{k,t,i}, (i = 0, \dots, \tau-1)$ with step size $\eta > 0$. After τ steps the client returns its local model $\theta_{k,t,\tau}$ (*i.e.*, the update $\Delta\theta_{k,t} = \theta_{k,t,\tau} - \theta_t$) to the server.

Global objective and aggregation. The federated goal is to maximize a weighted average of client objectives:

$$J_{\text{global}}(\theta) = \sum_{k=0}^{K-1} w_k\, J_k(\theta), \qquad w_k \geq 0, \sum_{k=0}^{K-1} w_k = 1. \tag{3}$$

In the FEDAGENT, the server uses uniform model averaging over the M participating clients each round (i.e., $w_k = \frac{1}{K}$ conceptually, with partial participation realized by S_t). Upon receiving the τ-step local models $\{\theta_{k,t,\tau}\}_{k \in S_t}$, the server performs model averaging: $\theta_{t+1} = \frac{1}{M} \sum_{k \in S_t} \theta_{k,t,\tau} = \theta_t + \frac{1}{M} \sum_{k \in S_t} \big(\theta_{k,t,\tau} - \theta_t\big)$. After T rounds the server outputs $\theta_{\text{final}} = \theta_T$.

3. FEDAGENTGYM: A Decentralized Agent Learning Environment

3.1. LLM Agents and Application Datasets

FEDAGENTGYM is designed as an environment to investigate the impact factors of training AI agents, especially LLM, in a decentralized way. It includes four types of LLM agents, including `Qwen2.5-{1.5,3,7}B-Instruct` and `Llama-3.2-3B-Instruct`, and two challenging application datasets (WebShop (Yao et al., 2022) and ALFWorld (Shridhar et al., 2021)), which require complex reasoning process and multi-step environment interactions. We adopt these two datasets to simulate the real-world scenarios where data privacy concerns are paramount.

WebShop is a web-based interactive platform that evaluates LLM agents within authentic e-commerce scenarios. Task completion requires agents to navigate a simulated HTML shopping interface to locate, browse, and purchase appropriate items. The dataset features an extensive catalog of over 1.1 million products paired with $12,000$ user instructions, creating a rich and varied action space.

ALFWorld provides an embodied simulation benchmark that evaluates LLM agents' capacity for sequential decision-making tasks. Each scenario presents the agent with a textual objective that must be achieved through iterative environment interaction. The dataset encompasses $3,827$ task instances spanning six types of household activities: Pick & Place (Pick), Examine in Light (Look), Clean & Place (Clean), Heat & Place (Heat), Cool & Place (Cool), and Pick Two & Place (Pick2).

3.2. Decentralized Settings

We comprehensively examine the impact of different decentralized settings on FEDAGENT performance across three critical dimensions. First, we vary **the number of samples per client**, which determines the sampling scope for each LLM agent's exploration of the action space and response generation, directly affecting both the diversity of experiences collected and the quality of policy gradient estimates. Second, we change **the number of clients selected per communication round**, controlling both the computational parallelism and the degree of heterogeneity in exploration strategies incorporated during global model aggregation. Third, we adjust **the number of local training batches per client per round**, governing the extent of local optimization on the sampled trajectories before synchronization with the central server. These parameters collectively influence fundamental trade-offs between exploration diversity, communication overhead, and convergence stability in the federated setting. Through extensive studies across these dimensions, we characterize how different decentralized training settings affect the final policy performance of FEDAGENT.

3.3. Heterogeneity Challenges

To systematically evaluate how FEDAGENT performs under realistic client distributions, we propose three novel and orthogonal heterogeneity definitions, as conventional heterogeneity dimensions in federated classification tasks (*e.g.*, feature or label skew) (Gao et al., 2022; Ye et al., 2024a) are not directly applicable. We also propose the corresponding client partitioning strategies, allowing us to understand the individual impact of different heterogeneity types.

Preference Heterogeneity: When Clients Have Different Task Preferences. In real-world federated learning, different clients often *prefer distinct types of tasks*. For example, in the ALFWorld, some users might frequently interact with kitchen-related tasks (like "put the apple in the fridge"), while others primarily encounter bedroom tasks (like "examine the lamp"). In WebShop, some may have mostly electronics searches while others mainly focus on clothing or home goods.

To simulate this preference heterogeneity, we propose the PREFERENCEPARTITION algorithm. The pseudo code is illustrated in Algorithm 2 in Appendix C.1. We model this by starting with the global distribution of task categories and introducing controlled noise to create client-specific preferences. Specifically, we add **Gaussian Noise** to the log-probabilities of the global category distribution, apply softmax normalization, and use the resulting probabilities to sample L instructions per client via **Multinomial Sampling**. As shown in Appendix C.2.1, this approach allows precise control over client distributions with a hyperparameter ω on topical preference heterogeneity, while maintaining the same total dataset size and per-client instruction count. More specifically, small noise values produce clients with similar task distributions, while larger noise creates highly specialized clients with distinct preferences.

Coverage Heterogeneity: When Clients Have Different Task Sampling Scopes. Even when clients encounter similar types of tasks, they may face vastly different quantities. A larger quantity of tasks indicates *coverage of a broader sampling scope per epoch* in reinforcement learning (we follow the setting in (Feng et al., 2025) to iteratively sample with replacement from the local data each epoch), while the sampling size remains fixed. Importantly, this differs from the quantity imbalance in conventional supervised federated classification tasks, where *training proceeds over the entire dataset each epoch*. In WebShop, for instance, some users might have extensive browsing histories with hundreds of product interactions, while others have only completed a few shopping sessions.

To model this heterogeneity, we develop the COVERAGEPARTITION algorithm. The pseudo code is shown in Algorithm 3 in Appendix C.1. We fix a global overlap target r (representing the average number of clients that

Method	ALFWorld							WebShop	
	Pick	Look	Clean	Heat	Cool	Pick2	All	Score	Succ.
Qwen2.5-1.5B-Instruct									
Local (Client 21)	42.9	25.0	38.5	37.5	14.3	14.3	29.7	69.9	57.0
Local (Client 42)	50.0	37.5	**76.9**	25.0	42.9	14.3	45.3	75.1	53.1
Local (Client 84)	50.0	37.5	46.2	25.0	28.6	0.0	34.4	72.7	47.7
Centralized	$64.3_{\pm 4.8}$	$37.5_{\pm 0.9}$	$69.2_{\pm 6.1}$	**50.0**$_{\pm 2.2}$	$42.9_{\pm 3.8}$	$28.6_{\pm 0.4}$	$51.6_{\pm 3.0}$	$79.9_{\pm 4.7}$	$57.8_{\pm 5.7}$
FEDAGENT	**80.0**$_{\pm 4.2}$	**75.0**$_{\pm 1.7}$	$53.8_{\pm 4.3}$	$37.5_{\pm 1.3}$	**83.3**$_{\pm 4.7}$	**50.0**$_{\pm 1.0}$	**64.1**$_{\pm 2.8}$	**83.2**$_{\pm 4.5}$	**61.7**$_{\pm 1.8}$
Qwen2.5-3B-Instruct									
Local (Client 21)	41.5	12.5	34.9	**51.0**	18.9	21.2	31.3	59.8	55.0
Local (Client 42)	46.5	37.5	24.4	15.0	33.7	33.3	28.2	61.3	59.3
Local (Client 84)	22.8	27.5	39.1	46.3	48.3	36.5	29.9	77.6	58.6
Centralized	$94.1_{\pm 0.9}$	**80.0**$_{\pm 2.5}$	**64.3**$_{\pm 1.4}$	$42.9_{\pm 2.6}$	$50.0_{\pm 2.7}$	$22.2_{\pm 5.2}$	$62.5_{\pm 4.2}$	**86.0**$_{\pm 1.5}$	**63.9**$_{\pm 2.8}$
FEDAGENT	**95.5**$_{\pm 4.3}$	$62.5_{\pm 3.0}$	$49.7_{\pm 1.7}$	$47.5_{\pm 2.4}$	**85.3**$_{\pm 3.6}$	**45.1**$_{\pm 2.1}$	**65.2**$_{\pm 3.9}$	$85.5_{\pm 3.4}$	$63.1_{\pm 3.1}$
Qwen2.5-7B-Instruct									
Local (Client 21)	35.5	25.0	61.0	25.9	35.8	**45.2**	38.4	70.9	49.2
Local (Client 42)	29.0	45.0	18.8	25.6	15.9	38.0	42.1	85.2	33.6
Local (Client 84)	34.7	47.5	44.4	51.3	40.1	21.8	35.7	60.6	39.3
Centralized	$93.7_{\pm 4.5}$	$82.5_{\pm 2.1}$	**71.5**$_{\pm 3.3}$	$47.9_{\pm 3.7}$	$63.2_{\pm 3.8}$	$31.9_{\pm 1.0}$	$73.3_{\pm 4.0}$	$78.8_{\pm 2.8}$	$64.7_{\pm 1.6}$
FEDAGENT	**94.5**$_{\pm 2.3}$	**85.0**$_{\pm 4.1}$	$56.0_{\pm 0.8}$	**62.5**$_{\pm 1.2}$	**86.7**$_{\pm 2.9}$	$42.8_{\pm 3.4}$	**75.5**$_{\pm 2.9}$	**89.0**$_{\pm 4.1}$	**68.9**$_{\pm 3.8}$
Llama-3.2-3B-Instruct									
Local (Client 21)	39.8	50.0	17.9	40.0	20.7	34.0	38.1	65.3	50.5
Local (Client 42)	18.2	55.0	41.9	34.3	41.0	25.0	35.0	67.0	51.0
Local (Client 84)	29.9	32.5	39.0	18.9	18.8	**37.6**	29.7	70.2	55.7
Centralized	$72.4_{\pm 4.6}$	**62.5**$_{\pm 4.5}$	$59.3_{\pm 3.1}$	$45.2_{\pm 0.5}$	$53.7_{\pm 2.2}$	$27.9_{\pm 3.0}$	$54.9_{\pm 2.9}$	**76.3**$_{\pm 3.7}$	$56.2_{\pm 1.6}$
FEDAGENT	**83.7**$_{\pm 1.7}$	$57.5_{\pm 6.0}$	**60.6**$_{\pm 3.4}$	**55.9**$_{\pm 0.9}$	**65.3**$_{\pm 2.8}$	$24.9_{\pm 3.1}$	**61.2**$_{\pm 3.3}$	$74.4_{\pm 4.9}$	**57.8**$_{\pm 3.2}$

Table 1: **Performance Comparison on ALFWorld and WebShop**. We report the averaged performance and the corresponding standard deviation for Centralized Training and FEDAGENT over three random seeds. For ALFWorld, the **Success Rate** (%) is reported for each subtask as well as for the overall dataset. For WebShop, both the **Task Score** (%) and the **Success Rate** (%) are reported.

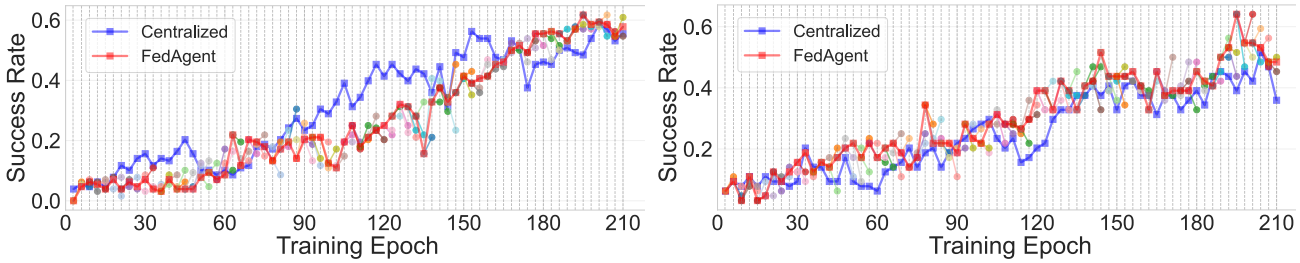
see each instruction) and draw each client's data quantity from a **Beta Distribution**, which we then map to the range $[L_{\min}, L_{\max}]$. Task instructions are allocated to clients using weighted sampling without replacement to satisfy both individual client quotas and the global overlap constraint. As shown in Appendix C.2.2, this method isolates the effect of task sampling scope on FEDAGENT performance while keeping the underlying task distribution consistent across clients. Also, this method controls the extent of coverage heterogeneity via hyperparameter ξ without impacting the overall mean of client quantities.

Hardness Heterogeneity: When Clients Face Different Task Difficulties. A particularly important but often overlooked source of heterogeneity is the overall difficulty of tasks that different clients encounter, which can be quantified by the *success rate* of tasks. For example, in ALFWorld, some clients might consistently face simple navigation tasks with high success rates, while others encounter complex multi-step reasoning tasks that frequently result in failure.

As demonstrated in Algorithm 4 in Appendix C.1, our proposed HARDNESSPARTITION algorithm addresses this by partitioning the task instruction pool into "successful" and "unsuccessful" examples with a pretrained checkpoint. Then, using our COVERAGEPARTITION method, we first distribute successful instructions according to a **Beta Distribution** that determines each client's success rate. We then fill remaining slots with unsuccessful examples sampled uniformly, ensuring all clients have exactly L instructions. As shown in Appendix C.2.3, this method enables us to study how different success rate distributions, which are controlled by a hyperparameter ξ' and measures the extent of hardness of task distributions for each client, affect FEDAGENT while maintaining consistent dataset sizes and global overlap patterns across all clients.

4. Is FedAgent Comparable to Centralized Agent Learning?

Experiment Setup. In this section, we aim to investigate the performance of FEDAGENT under a uniform client distribution, which is independent of the aforementioned three types of client heterogeneities. We partitioned the whole dataset (WebShop, ALFWorld) into 100 clients. Each client has 100 task instructions and there is a potential overlap between clients. 2 clients are randomly selected each round. Each client is trained for 3 epochs per round, with a total of



(a) WebShop (b) ALFWorld

Figure 2: **Training Dynamics of FEDAGENT and Centralized Training**. Circle marks with different colors indicate the model performance after training on *specific selected clients each round*. The red line refers to the performance of the *aggregated models on server* throughout the training process.

70 rounds and 210 epochs overall. For each epoch, 64 tasks are sampled iteratively with replacement from local data.

As for FEDAGENT, we adopt GRPO (Shao et al., 2024) for policy optimization. Then, following the literature in federated learning (Liu et al., 2024a), we select two typical baselines: **Centralized Agent Training** and **Local Agent Training**. Centralized Agent Training uses the full dataset (*i.e.*, 64 tasks are sampled iteratively from the whole dataset each epoch), while Local Agent Training uses only a specific client's dataset (we randomly selected one number 21 and obtained three client indexes 21, 42, and 84 as the baselines). Both of them run for the same total epochs as FEDAGENT and also adopt GRPO for policy optimization.

Result Analysis. As shown in Table 1, ❶ **FEDAGENT consistently outperforms Local Agent Training and achieves comparable performance to Centralized Agent Training**. For instance, on ALFWorld using `Qwen2.5-7B-Instruct`, FEDAGENT achieves a 75.5% total success rate compared to local training variants that range from 35.7% to 42.1%, while matching the centralized training performance of 73.3%. This pattern is consistently observed across different model scales (1.5B, 3B, 7B), model architectures (`qwen` and `llama`), and randomly selected client indexes (21, 42, and 84). Similarly, on the WebShop benchmark, FEDAGENT maintains this advantage with `Llama-3.2-3B-Instruct` achieving 57.8% success rate versus local training with different indexes ranging from 50.5% to 55.7%, while remaining competitive with centralized training at 56.2%. These results demonstrate the advantage of FEDAGENT in achieving competitive performance while preserving users' data privacy inherently.

Figure 2 shows the whole training dynamics of FEDAGENT and Centralized Agent Training with `Qwen2.5-1.5B-Instruct` on WebShop and ALFWorld datasets. ❷ **Both paradigms ultimately converge to similar success rates despite different training dynamics** ($\sim$ 0.6 for WebShop, $\sim$ 0.5 for ALFWorld). In WebShop (left), both approaches demonstrate steady monotonic improvement, with centralized training initially outperforming FEDAGENT until approximately epoch 120, after which both converge to similar success rates around 0.6. In contrast, ALFWorld (right) exhibits relatively more volatile training dynamics with frequent performance fluctuations for both methods, ultimately converging to success rates around 0.5. This further illustrates that FEDAGENT can achieve comparable performance with centralized training.

Insight 1 on the Effectiveness of FEDAGENT

FEDAGENT matches Centralized Agent Learning and outperforms Local Agent Learning: FEDAGENT incurs minimal performance penalty and has the benefits through diverse client contributions.

5. What is the impact of Different Decentralized Settings?

Experiment Setup. In this section, we aim to study the impact of different decentralized settings on FEDAGENT in FEDAGENTGYM by systematically varying three key hyperparameters across two different datasets (WebShop and ALFWorld). We adopt `Qwen2.5-1.5B-Instruct` for all configurations. The experimental setup examines: (1) **samples per client**. We test 100, 500, and 1,000 tasks per client to understand how task sampling scope affects FEDAGENT learning dynamics; (2) **clients selected per round**. We compare 1, 2, and 4 participating clients each round to analyze the effect of federation scale on performance; and (3) **epochs per client per round**. We evaluate 1, 3, and 5 local training epochs to determine the optimal number of local computations before aggregation. Since we keep the total number of epochs the same at 210 for all configurations, 1, 3, and 5 local training epochs correspond to 210, 70, and 42 total rounds, respectively.

Result Analysis. The results in Figure 3 demonstrate that ❸ **FEDAGENT exhibits distinct sensitivity patterns towards decentralized settings**. First, it shows notable sensitivity to the number of epochs per client per round. Moving

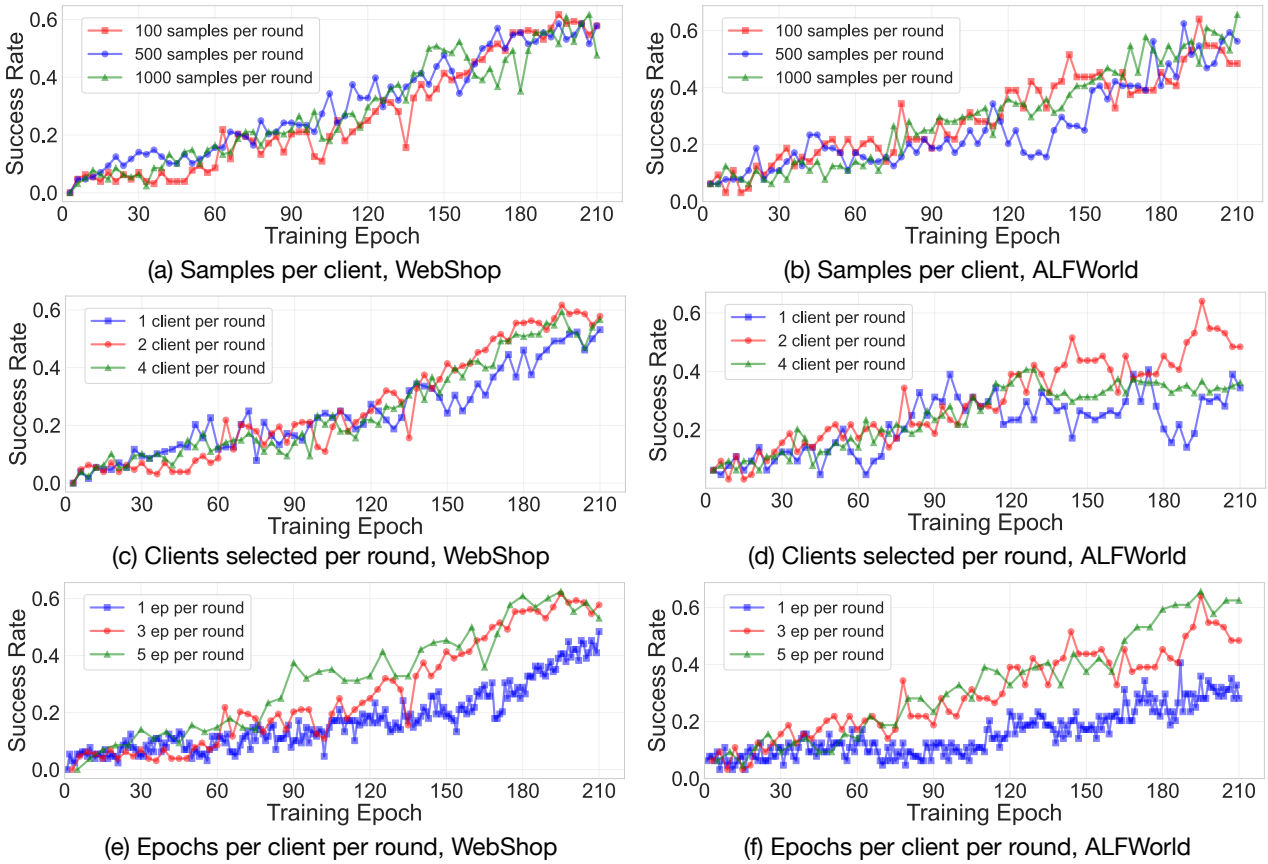


Figure 3: **Training Dynamics of FEDAGENT in Different Decentralized Settings**.

from 1 to 5 epochs per round leads to significant performance gains, especially after around 100 training epochs, highlighting that ❹ **shallow local updates may be insufficient to unlock the full potential of FEDAGENT**. On ALFWorld, FEDAGENT is also sensitive to the number of clients selected per round, with 2 clients per round outperforming 1 or 4, suggesting that ❺ **too few or too many clients could hinder convergence**. By contrast, FEDAGENT appears insensitive to the number of samples per client, as performance curves largely overlap across 100, 500, and 1,000 samples per round, suggesting that ❻ **the task sampling scope for one client beyond a certain threshold may not be the limiting factor**. Our studies offer valuable insights on the practical deployment of FEDAGENT and also suggest that ❼ **the specific decentralized configuration is important for FEDAGENT and may need to accommodate environments.**

Insight 2 on the Effectiveness of FEDAGENT

FEDAGENT's effectiveness depends on specific decentralized configurations: FEDAGENT exhibits relatively high sensitivity on clients selected per round and epochs per client per round, showing the importance of optimizing decentralized configurations.

6. What is the impact of Different Heterogeneity Challenges?

Experiment Setup. In this section, we aim to study the impact of different heterogeneity challenges on FEDAGENT in FEDAGENTGYM. As shown in Appendix C.2, we can leverage our proposed client partitioning strategies PREFERENCEPARTITION, COVERAGEPARTITION, and HARDNESSPARTITION to precisely control the extent of one form of heterogeneity (Preference, Coverage, or Hardness Heterogeneity) across clients with a hyperparameter ω, ξ, or ξ', respectively, without affecting the others. We keep the number of total epochs as 210 and the number of all clients as 100, which are consistent with the main experiments. We adopt `Qwen2.5-1.5B-Instruct` in the experiments.

Result Analysis. As shown in Figure 4, ❽ **FEDAGENT shows high robustness against the three heterogeneity challenges**. Across all scenarios, preference heterogeneity (panels a,b), coverage heterogeneity (panels c,d), and hardness heterogeneity (panels e,f), even when comparing relatively uniform settings (blue lines in Figure 4, $\omega = 0.1$, $\xi = 256$, $\xi' = 256$) against extremely high heterogeneity settings (green lines in Figure 4, $\omega = 0.9$, $\xi = 1$, $\xi' = 1$), FEDAGENT consistently achieves strong success rates that

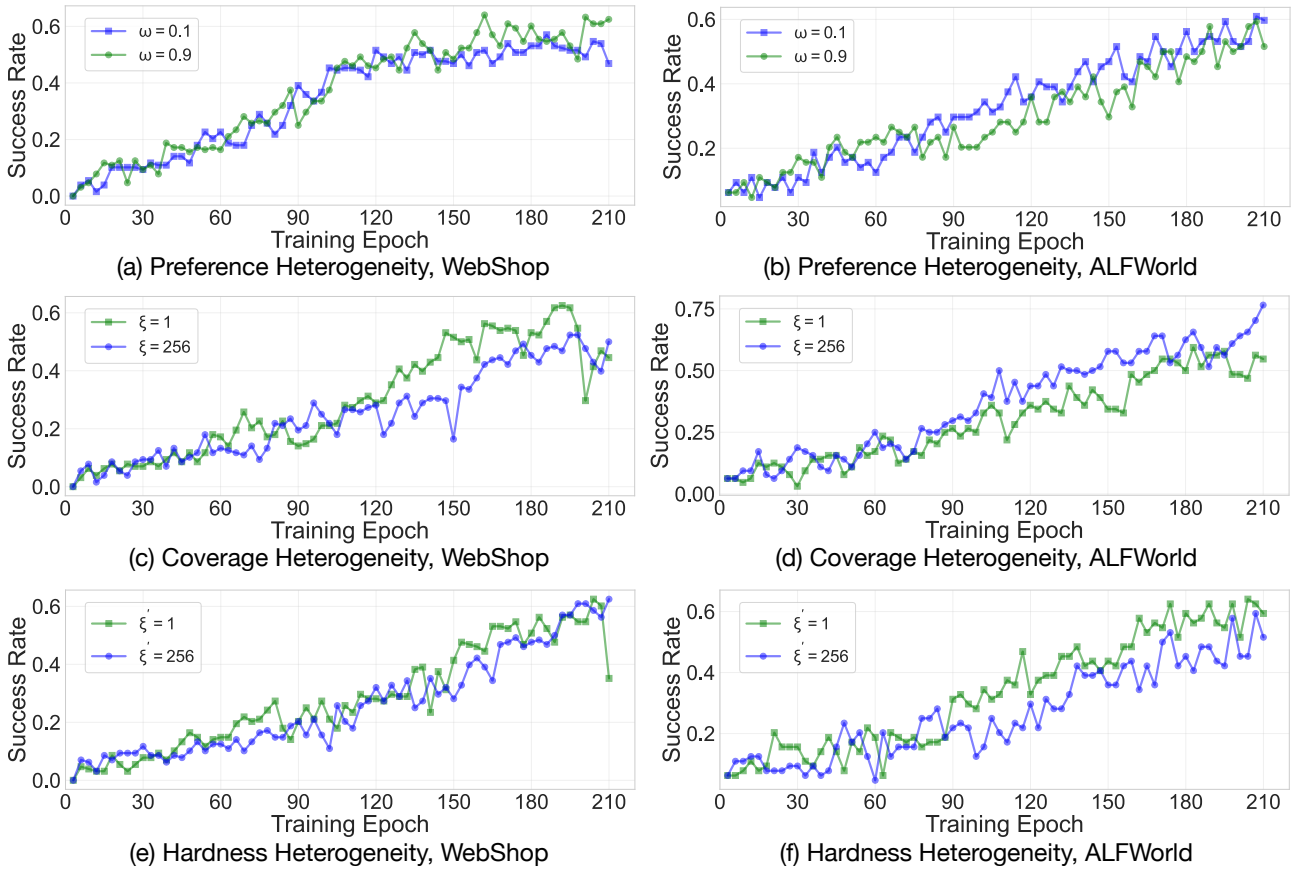


Figure 4: **Training Dynamics of FEDAGENT in Different Heterogeneity Challenges**.

steadily improve throughout training, The learning curves show that FEDAGENT maintains stable convergence behavior in both WebShop and ALFWorld environments regardless of heterogeneity intensity, with success rates generally reaching 0.5-0.6 by the end of training. Crucially, the performance degradation is minimal even under extreme heterogeneity conditions, indicating that ❾ **FEDAGENT has great potential to handle real-world scenarios across the full spectrum of heterogeneity challenges**.

Insight 3 on the Effectiveness of FEDAGENT

FEDAGENT shows high robustness against agent-specific heterogeneities: Under extreme conditions of Preference, Coverage, or Hardness Heterogeneity, FEDAGENT maintains stable convergence and achieves consistent final performance.

7. Related Work

RL has been instrumental in empowering LLM agents to function effectively in dynamic and open-ended environments. Initial studies leveraged traditional RL approaches like DQN (Mnih et al., 2015) for training LLM agents in text-based gaming environments (Narasimhan et al., 2015). Subsequent research began incorporating value-based techniques across broader agent applications such as Android device manipulation (Rawles et al., 2023) and embodied environments like ALFWorld (Shridhar et al., 2021). Contemporary methods have expanded RL training to encompass sophisticated web-based and application-specific tasks (Zhou et al., 2024c; Putta et al., 2024). In previous works, real-world task queries and trajectories have been essential for training AI agents in practical applications. However, they are becoming increasingly difficult to acquire due to privacy concerns. Our work makes an initial effort to explore training AI agents without compromising user data privacy.

8. Conclusion

In this work, we explored FEDAGENT (Federated Agent Reinforcement Learning), a new collaborative paradigm to train AI agent, particularly LLMs, across distributed clients, and built FEDAGENTGYM, the first decentralized agent learning environment. Extensive empirical studies demonstrate that FEDAGENT can achieve performance on par with centralized training and maintain strong robustness to heterogeneities. Our work validates the feasibility of training AI agents while protecting user data privacy and charts new research directions in agent learning.

Impact Statement

This work explores federated agent reinforcement learning, a paradigm designed to train AI agents collaboratively while preserving user privacy. Our work addresses a critical barrier to AI agent deployment: the tension between data-hungry training requirements and user privacy expectations. By enabling effective agent learning without centralized data collection, FEDAGENT could democratize access to high-quality AI agents for individuals and organizations that cannot share sensitive interaction data due to regulatory constraints (e.g., GDPR, HIPAA) or competitive concerns. This is particularly relevant for domains such as personal assistants, healthcare navigation, and financial planning, where user interactions contain sensitive information. Furthermore, federated learning inherently distributes computational costs across participants, potentially reducing the environmental footprint compared to centralized training on GPU clusters.

References

Acar, D. A. E., Zhao, Y., Navarro, R. M., Mattina, M., Whatmough, P. N., and Saligrama, V. Federated learning based on dynamic regularization. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net, 2021. URL https://openreview.net/forum?id=B7v4QMR6Z9w.

Ahmadian, A., Cremer, C., Gallé, M., Fadaee, M., Kreutzer, J., Pietquin, O., Üstün, A., and Hooker, S. Back to basics: Revisiting reinforce-style optimization for learning from human feedback in llms. In Ku, L., Martins, A., and Srikumar, V. (eds.), *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2024, Bangkok, Thailand, August 11-16, 2024*, pp. 12248–12267. Association for Computational Linguistics, 2024. doi: 10.18653/V1/2024.ACL-LONG.662. URL https://doi.org/10.18653/v1/2024.acl-long.662.

Alistarh, D., Grubic, D., Li, J., Tomioka, R., and Vojnovic, M. QSGD: communication-efficient SGD via gradient quantization and encoding. In Guyon, I., von Luxburg, U., Bengio, S., Wallach, H. M., Fergus, R., Vishwanathan, S. V. N., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pp. 1709–1720, 2017. URL https://proceedings.neurips.cc/paper/2017/hash/6c340f25839e6acdc73414517203f5f0-Abstract.html.

Arivazhagan, M. G., Aggarwal, V., Singh, A. K., and Choudhary, S. Federated learning with personalization layers. *CoRR*, abs/1912.00818, 2019. URL http://arxiv.org/abs/1912.00818.

Bai, H., Zhou, Y., Pan, J., Cemri, M., Suhr, A., Levine, S., and Kumar, A. Digirl: Training in-the-wild device-control agents with autonomous reinforcement learning. In Globersons, A., Mackey, L., Belgrave, D., Fan, A., Paquet, U., Tomczak, J. M., and Zhang, C. (eds.), *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*, 2024a. URL http://papers.nips.cc/paper_files/paper/2024/hash/1704ddd0bb89f159dfe609b32c889995-Abstract-Conference.html.

Bai, J., Chen, D., Qian, B., Yao, L., and Li, Y. Federated fine-tuning of large language models under heterogeneous tasks and client resources. In Globersons, A., Mackey, L., Belgrave, D., Fan, A., Paquet, U., Tomczak, J. M., and Zhang, C. (eds.), *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*, 2024b. URL http://papers.nips.cc/paper_files/paper/2024/hash/1a134b50202088aa8c595cc99b310e5a-Abstract-Conference.html.

Bian, J., Wang, L., Zhang, L., and Xu, J. Fedalt: Federated fine-tuning through adaptive local training with rest-of-the-world lora. *CoRR*, abs/2503.11880, 2025. doi: 10.48550/ARXIV.2503.11880. URL https://doi.org/10.48550/arXiv.2503.11880.

Chen, B., Shu, C., Shareghi, E., Collier, N., Narasimhan, K., and Yao, S. Fireact: Toward language agent fine-tuning. *CoRR*, abs/2310.05915, 2023. doi: 10.48550/ARXIV.2310.05915. URL https://doi.org/10.48550/arXiv.2310.05915.

Chen, K., Cusumano-Towner, M. F., Huval, B., Petrenko, A., Hamburger, J., Koltun, V., and Krähenbühl, P. Reinforcement learning for long-horizon interactive LLM agents. *CoRR*, abs/2502.01600, 2025a. doi: 10.48550/ARXIV.2502.01600. URL https://doi.org/10.48550/arXiv.2502.01600.

Chen, S., Zhu, T., Wang, Z., Zhang, J., Wang, K., Gao, S., Xiao, T., Teh, Y. W., He, J., and Li, M. Internalizing world models via self-play finetuning for agentic RL. *CoRR*, abs/2510.15047, 2025b. doi: 10.48550/ARXIV.2510.15047. URL https://doi.org/10.48550/arXiv.2510.15047.

Chen, X., Shi, Y., Lan, Q., Qiu, Y., and Gu, X. Fed-se: Federated self-evolution for privacy-constrained multi-

environment LLM agents. *CoRR*, abs/2512.08870, 2025c. doi: 10.48550/ARXIV.2512.08870. URL https://doi.org/10.48550/arXiv.2512.08870.

Chen, Z., Cui, H., Wu, E., and Xi, Y. Efficient adaptive federated optimization of federated learning for iot. *CoRR*, abs/2206.11448, 2022. doi: 10.48550/ARXIV.2206.11448. URL https://doi.org/10.48550/arXiv.2206.11448.

Chen, Z., Zhao, Z., Zhang, K., Liu, B., Qi, Q., Wu, Y., Kalluri, T., Cao, S., Xiong, Y., Tong, H., Yao, H., Li, H., Zhu, J., Li, X., Song, D., Li, B., Weston, J., and Huynh, D. Scaling agent learning via experience synthesis. *CoRR*, abs/2511.03773, 2025d. doi: 10.48550/ARXIV.2511.03773. URL https://doi.org/10.48550/arXiv.2511.03773.

Chen, Z. et al. Offline federated deep reinforcement learning with awareness of policy inconsistency. *arXiv preprint*, 2025e.

Cheng, M., Ouyang, J., Yu, S., Yan, R., Luo, Y., Liu, Z., Wang, D., Liu, Q., and Chen, E. Agent-r1: Training powerful LLM agents with end-to-end reinforcement learning. *CoRR*, abs/2511.14460, 2025. doi: 10.48550/ARXIV.2511.14460. URL https://doi.org/10.48550/arXiv.2511.14460.

Cheruiyot, K., Kiprotich, N., Kungurtsev, V., Mugo, K., Mwirigi, V., and Ngesa, M. A survey of multi agent reinforcement learning: Federated learning and cooperative and noncooperative decentralized regimes. *CoRR*, abs/2507.06278, 2025. doi: 10.48550/ARXIV.2507.06278. URL https://doi.org/10.48550/arXiv.2507.06278.

Cho, Y. J., Liu, L., Xu, Z., Fahrezi, A., and Joshi, G. Heterogeneous lora for federated fine-tuning of on-device foundation models. In Al-Onaizan, Y., Bansal, M., and Chen, Y. (eds.), *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing, EMNLP 2024, Miami, FL, USA, November 12-16, 2024*, pp. 12903–12913. Association for Computational Linguistics, 2024. doi: 10.18653/V1/2024.EMNLP-MAIN.717. URL https://doi.org/10.18653/v1/2024.emnlp-main.717.

Collins, L., Hassani, H., Mokhtari, A., and Shakkottai, S. Exploiting shared representations for personalized federated learning. In Meila, M. and Zhang, T. (eds.), *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*, volume 139 of *Proceedings of Machine Learning Research*, pp. 2089–2099. PMLR, 2021. URL http://proceedings.mlr.press/v139/collins21a.html.

DeepSeek-AI, Guo, D., Yang, D., Zhang, H., Song, J., Zhang, R., Xu, R., Zhu, Q., Ma, S., Wang, P., Bi, X., Zhang, X., Yu, X., Wu, Y., Wu, Z. F., Gou, Z., Shao, Z., Li, Z., Gao, Z., Liu, A., Xue, B., Wang, B., Wu, B., Feng, B., Lu, C., Zhao, C., Deng, C., Zhang, C., Ruan, C., Dai, D., Chen, D., Ji, D., Li, E., Lin, F., Dai, F., Luo, F., Hao, G., Chen, G., Li, G., Zhang, H., Bao, H., Xu, H., Wang, H., Ding, H., Xin, H., Gao, H., Qu, H., Li, H., Guo, J., Li, J., Wang, J., Chen, J., Yuan, J., Qiu, J., Li, J., Cai, J. L., Ni, J., Liang, J., Chen, J., Dong, K., Hu, K., Gao, K., Guan, K., Huang, K., Yu, K., Wang, L., Zhang, L., Zhao, L., Wang, L., Zhang, L., Xu, L., Xia, L., Zhang, M., Zhang, M., Tang, M., Li, M., Wang, M., Li, M., Tian, N., Huang, P., Zhang, P., Wang, Q., Chen, Q., Du, Q., Ge, R., Zhang, R., Pan, R., Wang, R., Chen, R. J., Jin, R. L., Chen, R., Lu, S., Zhou, S., Chen, S., Ye, S., Wang, S., Yu, S., Zhou, S., Pan, S., Li, S. S., Zhou, S., Wu, S., Ye, S., Yun, T., Pei, T., Sun, T., Wang, T., Zeng, W., Zhao, W., Liu, W., Liang, W., Gao, W., Yu, W., Zhang, W., Xiao, W. L., An, W., Liu, X., Wang, X., Chen, X., Nie, X., Cheng, X., Liu, X., Xie, X., Liu, X., Yang, X., Li, X., Su, X., Lin, X., Li, X. Q., Jin, X., Shen, X., Chen, X., Sun, X., Wang, X., Song, X., Zhou, X., Wang, X., Shan, X., Li, Y. K., Wang, Y. Q., Wei, Y. X., Zhang, Y., Xu, Y., Li, Y., Zhao, Y., Sun, Y., Wang, Y., Yu, Y., Zhang, Y., Shi, Y., Xiong, Y., He, Y., Piao, Y., Wang, Y., Tan, Y., Ma, Y., Liu, Y., Guo, Y., Ou, Y., Wang, Y., Gong, Y., Zou, Y., He, Y., Xiong, Y., Luo, Y., You, Y., Liu, Y., Zhou, Y., Zhu, Y. X., Xu, Y., Huang, Y., Li, Y., Zheng, Y., Zhu, Y., Ma, Y., Tang, Y., Zha, Y., Yan, Y., Ren, Z. Z., Ren, Z., Sha, Z., Fu, Z., Xu, Z., Xie, Z., Zhang, Z., Hao, Z., Ma, Z., Yan, Z., Wu, Z., Gu, Z., Zhu, Z., Liu, Z., Li, Z., Xie, Z., Song, Z., Pan, Z., Huang, Z., Xu, Z., Zhang, Z., and Zhang, Z. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv: 2501.12948*, 2025.

Deng, Y., Kamani, M. M., and Mahdavi, M. Adaptive personalized federated learning. *arXiv preprint arXiv:2003.13461*, 2020.

Ding, H., Liu, P., Wang, J., Ji, Z., Cao, M., Zhang, R., Ai, L., Yang, E., Shi, T., and Yu, L. Dynaweb: Model-based reinforcement learning of web agents. *CoRR*, abs/2601.22149, 2026. doi: 10.48550/ARXIV.2601.22149. URL https://doi.org/10.48550/arXiv.2601.22149.

Dinh, C. T., Tran, N. H., and Nguyen, T. D. Personalized federated learning with moreau envelopes. In Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M., and Lin, H. (eds.), *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020. URL https:

//proceedings.neurips.cc/paper/2020/hash/f4f1f13c8289ac1b1ee0ff176b56fc60-Abstract.html.

Du, Y., Ye, R., Yuchi, F., Zhao, W., Qu, J., Wang, Y., and Chen, S. Feddqc: Data quality control in federated instruction-tuning of large language models. In Che, W., Nabende, J., Shutova, E., and Pilehvar, M. T. (eds.), *Findings of the Association for Computational Linguistics, ACL 2025, Vienna, Austria, July 27 - August 1, 2025*, volume ACL 2025 of *Findings of ACL*, pp. 15267–15291. Association for Computational Linguistics, 2025. URL https://aclanthology.org/2025.findings-acl.791/.

Fallah, A., Mokhtari, A., and Ozdaglar, A. E. Personalized federated learning with theoretical guarantees: A model-agnostic meta-learning approach. In Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M., and Lin, H. (eds.), *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020. URL https://proceedings.neurips.cc/paper/2020/hash/24389bfe4fe2eba8bf9aa9203a44cdad-Abstract.html.

Fan, F. X., Ma, Y., Dai, Z., Jing, W., Tan, C., and Low, B. K. H. Fault-tolerant federated reinforcement learning with theoretical guarantee. In Ranzato, M., Beygelzimer, A., Dauphin, Y. N., Liang, P., and Vaughan, J. W. (eds.), *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, pp. 1007–1021, 2021. URL https://proceedings.neurips.cc/paper/2021/hash/080acdcce72c06873a773c4311c2e464-Abstract.html.

Fan, F. X., Tan, C., Ong, Y.-S., Wattenhofer, R., and Ooi, W.-T. Fedrlhf: A convergence-guaranteed federated framework for privacy-preserving and personalized rlhf. In *Proceedings of the 24th International Conference on Autonomous Agents and Multiagent Systems*, AAMAS'25, pp. 713–721, Richland, SC, 2025. International Foundation for Autonomous Agents and Multiagent Systems. ISBN 9798400714269.

Feng, L., Xue, Z., Liu, T., and An, B. Group-in-group policy optimization for LLM agent training. *CoRR*, abs/2505.10978, 2025. doi: 10.48550/ARXIV.2505.10978. URL https://doi.org/10.48550/arXiv.2505.10978.

Fu, D., He, K., Wang, Y., Hong, W., Gongque, Z., Zeng, W., Wang, W., Wang, J., Cai, X., and Xu, W. Agentrefine: Enhancing agent generalization through refinement tuning. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net, 2025. URL https://openreview.net/forum?id=FDimWzmcWn.

Gao, D., Yao, X., and Yang, Q. A survey on heterogeneous federated learning. *CoRR*, abs/2210.04505, 2022. doi: 10.48550/ARXIV.2210.04505. URL https://doi.org/10.48550/arXiv.2210.04505.

Gao, H., Geng, J., Hua, W., Hu, M., Juan, X., Liu, H., Liu, S., Qiu, J., Qi, X., Wu, Y., Wang, H., Xiao, H., Zhou, Y., Zhang, S., Zhang, J., Xiang, J., Fang, Y., Zhao, Q., Liu, D., Ren, Q., Qian, C., Wang, Z., Hu, M., Wang, H., Wu, Q., Ji, H., and Wang, M. A survey of self-evolving agents: On path to artificial super intelligence. *CoRR*, abs/2507.21046, 2025. doi: 10.48550/ARXIV.2507.21046. URL https://doi.org/10.48550/arXiv.2507.21046.

Geyer, R. C., Klein, T., and Nabi, M. Differentially private federated learning: A client level perspective. *CoRR*, abs/1712.07557, 2017. URL http://arxiv.org/abs/1712.07557.

Golubev, A., Trofimova, M., Polezhaev, S., Badertdinov, I., Nekrashevich, M., Shevtsov, A., Karasik, S., Abramov, S., Andriushchenko, A., Fisin, F., Skvortsov, S., and Yangel, B. Training long-context, multi-turn software engineering agents with reinforcement learning. *CoRR*, abs/2508.03501, 2025. doi: 10.48550/ARXIV.2508.03501. URL https://doi.org/10.48550/arXiv.2508.03501.

Guo, P., Zeng, S., Wang, Y., Fan, H., Wang, F., and Qu, L. Selective aggregation for low-rank adaptation in federated learning. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net, 2025a. URL https://openreview.net/forum?id=iX3uESGdsO.

Guo, Y., Tang, X., and Lin, T. Enhancing clustered federated learning: Integration of strategies and improved methodologies. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net, 2025b. URL https://openreview.net/forum?id=zPDpdk3V8L.

Hsu, T. H., Qi, H., and Brown, M. Measuring the effects of non-identical data distribution for federated visual classification. *CoRR*, abs/1909.06335, 2019. URL http://arxiv.org/abs/1909.06335.

Huang, G. and Shu, T. Federated oriented learning: A practical one-shot personalized federated learning framework. In Singh, A., Fazel, M., Hsu, D., Lacoste-Julien, S., Berkenkamp, F., Maharaj, T., Wagstaff, K.,

and Zhu, J. (eds.), *Forty-second International Conference on Machine Learning, ICML 2025, Vancouver, BC, Canada, July 13-19, 2025*, volume 267 of *Proceedings of Machine Learning Research*. PMLR / OpenReview.net, 2025. URL https://proceedings.mlr.press/v267/huang25ae.html.

Hwang, U. and Hong, S. Federated reinforcement learning in heterogeneous environments. *CoRR*, abs/2507.14487, 2025. doi: 10.48550/ARXIV.2507.14487. URL https://doi.org/10.48550/arXiv.2507.14487.

Hyeon-Woo, N., Ye-Bin, M., and Oh, T. Fedpara: Low-rank hadamard product for communication-efficient federated learning. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net, 2022. URL https://openreview.net/forum?id=d71n4ftoCBy.

Jimenez, C. E., Yang, J., Wettig, A., Yao, S., Pei, K., Press, O., and Narasimhan, K. R. Swe-bench: Can language models resolve real-world github issues? In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net, 2024. URL https://openreview.net/forum?id=VTF8yNQM66.

Jin, B., Zeng, H., Yue, Z., Wang, D., Zamani, H., and Han, J. Search-r1: Training llms to reason and leverage search engines with reinforcement learning. *CoRR*, abs/2503.09516, 2025. doi: 10.48550/ARXIV.2503.09516. URL https://doi.org/10.48550/arXiv.2503.09516.

Jin, H., Peng, Y., Yang, W., Wang, S., and Zhang, Z. Federated reinforcement learning with environment heterogeneity. In Camps-Valls, G., Ruiz, F. J. R., and Valera, I. (eds.), *International Conference on Artificial Intelligence and Statistics, AISTATS 2022, 28-30 March 2022, Virtual Event*, volume 151 of *Proceedings of Machine Learning Research*, pp. 18–37. PMLR, 2022. URL https://proceedings.mlr.press/v151/jin22a.html.

Kairouz, P., McMahan, H. B., Avent, B., Bellet, A., Bennis, M., Bhagoji, A. N., Bonawitz, K. A., Charles, Z., Cormode, G., Cummings, R., D'Oliveira, R. G. L., Eichner, H., Rouayheb, S. E., Evans, D., Gardner, J., Garrett, Z., Gascón, A., Ghazi, B., Gibbons, P. B., Gruteser, M., Harchaoui, Z., He, C., He, L., Huo, Z., Hutchinson, B., Hsu, J., Jaggi, M., Javidi, T., Joshi, G., Khodak, M., Konečný, J., Korolova, A., Koushanfar, F., Koyejo, S., Lepoint, T., Liu, Y., Mittal, P., Mohri, M., Nock, R., Özgür, A., Pagh, R., Qi, H., Ramage, D., Raskar, R., Raykova, M., Song, D., Song, W., Stich, S. U., Sun, Z., Suresh, A. T., Tramèr, F., Vepakomma, P., Wang, J., Xiong, L., Xu, Z., Yang, Q., Yu, F. X., Yu, H., and Zhao, S. Advances and open problems in federated learning. *Found. Trends Mach. Learn.*, 14(1-2):1–210, 2021. doi: 10.1561/2200000083. URL https://doi.org/10.1561/2200000083.

Karimireddy, S. P., Kale, S., Mohri, M., Reddi, S. J., Stich, S. U., and Suresh, A. T. SCAFFOLD: stochastic controlled averaging for federated learning. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, volume 119 of *Proceedings of Machine Learning Research*, pp. 5132–5143. PMLR, 2020. URL http://proceedings.mlr.press/v119/karimireddy20a.html.

Khodadadian, S., Sharma, P., Joshi, G., and Maguluri, S. T. Federated reinforcement learning: Linear speedup under markovian sampling. In Chaudhuri, K., Jegelka, S., Song, L., Szepesvári, C., Niu, G., and Sabato, S. (eds.), *International Conference on Machine Learning, ICML 2022, 17-23 July 2022, Baltimore, Maryland, USA*, volume 162 of *Proceedings of Machine Learning Research*, pp. 10997–11057. PMLR, 2022. URL https://proceedings.mlr.press/v162/khodadadian22a.html.

Konečný, J., McMahan, H. B., Yu, F. X., Richtárik, P., Suresh, A. T., and Bacon, D. Federated learning: Strategies for improving communication efficiency. *CoRR*, abs/1610.05492, 2016. URL http://arxiv.org/abs/1610.05492.

Kuang, W., Qian, B., Li, Z., Chen, D., Gao, D., Pan, X., Xie, Y., Li, Y., Ding, B., and Zhou, J. Federatedscope-llm: A comprehensive package for fine-tuning large language models in federated learning. In Baeza-Yates, R. and Bonchi, F. (eds.), *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, KDD 2024, Barcelona, Spain, August 25-29, 2024*, pp. 5260–5271. ACM, 2024. doi: 10.1145/3637528.3671573. URL https://doi.org/10.1145/3637528.3671573.

Labbi, S., Tiapkin, D., Mancini, L., Mangold, P., and Moulines, E. Federated UCBVI: communication-efficient federated regret minimization with heterogeneous agents. In Li, Y., Mandt, S., Agrawal, S., and Khan, M. E. (eds.), *International Conference on Artificial Intelligence and Statistics, AISTATS 2025, Mai Khao, Thailand, 3-5 May 2025*, volume 258 of *Proceedings of Machine Learning Research*, pp. 1315–1323. PMLR, 2025. URL https://proceedings.mlr.press/v258/labbi25a.html.

Lan, G., Han, D., Hashemi, A., Aggarwal, V., and Brinton, C. Asynchronous federated reinforcement learning with policy gradient updates: Algorithm design and convergence analysis. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net, 2025. URL https://openreview.net/forum?id=5DUekOKWcS.

Li, D. and Wang, J. Fedmd: Heterogenous federated learning via model distillation. *CoRR*, abs/1910.03581, 2019. URL http://arxiv.org/abs/1910.03581.

Li, Q., He, B., and Song, D. Model-contrastive federated learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10713–10722, 2021a.

Li, T., Sahu, A. K., Talwalkar, A., and Smith, V. Federated learning: Challenges, methods, and future directions. *IEEE Signal Process. Mag.*, 37(3):50–60, 2020a. doi: 10.1109/MSP.2020.2975749. URL https://doi.org/10.1109/MSP.2020.2975749.

Li, T., Sahu, A. K., Zaheer, M., Sanjabi, M., Talwalkar, A., and Smith, V. Federated optimization in heterogeneous networks. In Dhillon, I. S., Papailiopoulos, D. S., and Sze, V. (eds.), *Proceedings of the Third Conference on Machine Learning and Systems, MLSys 2020, Austin, TX, USA, March 2-4, 2020*. mlsys.org, 2020b. URL https://proceedings.mlsys.org/paper_files/paper/2020/hash/1f5fe83998a09396ebe6477d9475ba0c-Abstract.html.

Li, T., Hu, S., Beirami, A., and Smith, V. Ditto: Fair and robust federated learning through personalization. In Meila, M. and Zhang, T. (eds.), *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*, volume 139 of *Proceedings of Machine Learning Research*, pp. 6357–6368. PMLR, 2021b. URL http://proceedings.mlr.press/v139/li21h.html.

Li, X., Huang, K., Yang, W., Wang, S., and Zhang, Z. On the convergence of fedavg on non-iid data. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net, 2020c. URL https://openreview.net/forum?id=HJxNAnVtDS.

Li, X., Jiang, M., Zhang, X., Kamp, M., and Dou, Q. Fedbn: Federated learning on non-iid features via local batch normalization. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net, 2021c. URL https://openreview.net/forum?id=6YEQUn0QICG.

Li, Z., Long, G., and Zhou, T. Federated recommendation with additive personalization. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net, 2024. URL https://openreview.net/forum?id=xkXdE81mOK.

Liang, X., Liu, Y., Chen, T., Liu, M., and Yang, Q. Federated transfer reinforcement learning for autonomous driving. *CoRR*, abs/1910.06001, 2019. URL http://arxiv.org/abs/1910.06001.

Liao, Y., Huang, W., Wan, G., Liang, J., Yang, B., and Ye, M. Splitting with importance-aware updating for heterogeneous federated learning with large language models. In Singh, A., Fazel, M., Hsu, D., Lacoste-Julien, S., Berkenkamp, F., Maharaj, T., Wagstaff, K., and Zhu, J. (eds.), *Forty-second International Conference on Machine Learning, ICML 2025, Vancouver, BC, Canada, July 13-19, 2025*, volume 267 of *Proceedings of Machine Learning Research*. PMLR / OpenReview.net, 2025. URL https://proceedings.mlr.press/v267/liao25c.html.

Lightman, H., Kosaraju, V., Burda, Y., Edwards, H., Baker, B., Lee, T., Leike, J., Schulman, J., Sutskever, I., and Cobbe, K. Let's verify step by step. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net, 2024. URL https://openreview.net/forum?id=v8L0pN6EOi.

Lin, T., Kong, L., Stich, S. U., and Jaggi, M. Ensemble distillation for robust model fusion in federated learning. In Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M., and Lin, H. (eds.), *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020. URL https://proceedings.neurips.cc/paper/2020/hash/18df51b97ccd68128e994804f3eccc87-Abstract.html.

Lin, Y., Han, S., Mao, H., Wang, Y., and Dally, B. Deep gradient compression: Reducing the communication bandwidth for distributed training. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net, 2018. URL https://openreview.net/forum?id=SkhQHMW0W.

Ling, Z., Chen, D., Yao, L., Li, Y., and Shen, Y. FedMeZO: On the convergence of zeroth-order federated tuning for large language models. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2024.

Liu, B., Wang, L., and Liu, M. Lifelong federated reinforcement learning for cloud robotic systems. *IEEE Access*, 7: 170011–170022, 2019.

Liu, B., Lv, N., Guo, Y., and Li, Y. Recent advances on federated learning: A systematic survey. *Neurocomputing*, 597:128019, 2024a. doi: 10.1016/J.NEUCOM.2024.

128019. URL https://doi.org/10.1016/j.neucom.2024.128019.

Liu, B., Li, X., Zhang, J., Wang, J., He, T., Hong, S., Liu, H., Zhang, S., Song, K., Zhu, K., Cheng, Y., Wang, S., Wang, X., Luo, Y., Jin, H., Zhang, P., Liu, O., Chen, J., Zhang, H., Yu, Z., Shi, H., Li, B., Wu, D., Teng, F., Jia, X., Xu, J., Xiang, J., Lin, Y., Liu, T., Liu, T., Su, Y., Sun, H., Berseth, G., Nie, J., Foster, I. T., Ward, L. T., Wu, Q., Gu, Y., Zhuge, M., Tang, X., Wang, H., You, J., Wang, C., Pei, J., Yang, Q., Qi, X., and Wu, C. Advances and challenges in foundation agents: From brain-inspired intelligence to evolutionary, collaborative, and safe systems. *CoRR*, abs/2504.01990, 2025a. doi: 10.48550/ARXIV.2504.01990. URL https://doi.org/10.48550/arXiv.2504.01990.

Liu, H., Wen, R., Nair, S., Liu, J., Lou, W., Zhang, C., Yeoh, W., Vorobeychik, Y., and Zhang, N. Ecolora: Communication-efficient federated fine-tuning of large language models. In Christodoulopoulos, C., Chakraborty, T., Rose, C., and Peng, V. (eds.), *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing, EMNLP 2025, Suzhou, China, November 4-9, 2025*, pp. 20732–20746. Association for Computational Linguistics, 2025b. doi: 10.18653/V1/2025.EMNLP-MAIN.1046. URL https://doi.org/10.18653/v1/2025.emnlp-main.1046.

Liu, S., Liang, Z., Lyu, X., and Amato, C. LLM collaboration with multi-agent reinforcement learning. *CoRR*, abs/2508.04652, 2025c. doi: 10.48550/ARXIV.2508.04652. URL https://doi.org/10.48550/arXiv.2508.04652.

Liu, X., Yu, H., Zhang, H., Xu, Y., Lei, X., Lai, H., Gu, Y., Ding, H., Men, K., Yang, K., Zhang, S., Deng, X., Zeng, A., Du, Z., Zhang, C., Shen, S., Zhang, T., Su, Y., Sun, H., Huang, M., Dong, Y., and Tang, J. Agentbench: Evaluating llms as agents. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net, 2024b. URL https://openreview.net/forum?id=zAdUB0aCTQ.

Liu, X., Wang, K., Wu, Y., Huang, F., Li, Y., Zhang, J., and Jiao, J. Agentic reinforcement learning with implicit step rewards. *CoRR*, abs/2509.19199, 2025d. doi: 10.48550/ARXIV.2509.19199. URL https://doi.org/10.48550/arXiv.2509.19199.

Liu, Z., Chai, J., Zhu, X., Tang, S., Ye, R., Zhang, B., Bai, L., and Chen, S. Ml-agent: Reinforcing LLM agents for autonomous machine learning engineering. *CoRR*, abs/2505.23723, 2025e. doi: 10.48550/ARXIV.2505.23723. URL https://doi.org/10.48550/arXiv.2505.23723.

Luo, M., Jain, N., Singh, J., Tan, S., et al. DeepSWE: Training a fully open-sourced, state-of-the-art coding agent by scaling RL. *Technical Report, Together AI*, 2025.

McMahan, B., Moore, E., Ramage, D., Hampson, S., and y Arcas, B. A. Communication-efficient learning of deep networks from decentralized data. In Singh, A. and Zhu, X. J. (eds.), *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics, AISTATS 2017, 20-22 April 2017, Fort Lauderdale, FL, USA*, volume 54 of *Proceedings of Machine Learning Research*, pp. 1273–1282. PMLR, 2017. URL http://proceedings.mlr.press/v54/mcmahan17a.html.

McMahan, H. B., Ramage, D., Talwar, K., and Zhang, L. Learning differentially private recurrent language models. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net, 2018. URL https://openreview.net/forum?id=BJ0hF1Z0b.

Mialon, G., Fourrier, C., Wolf, T., LeCun, Y., and Scialom, T. GAIA: a benchmark for general AI assistants. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net, 2024. URL https://openreview.net/forum?id=fibxvahvs3.

Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A. A., Veness, J., Bellemare, M. G., Graves, A., Riedmiller, M. A., Fidjeland, A., Ostrovski, G., Petersen, S., Beattie, C., Sadik, A., Antonoglou, I., King, H., Kumaran, D., Wierstra, D., Legg, S., and Hassabis, D. Human-level control through deep reinforcement learning. *Nat.*, 518(7540): 529–533, 2015. doi: 10.1038/NATURE14236. URL https://doi.org/10.1038/nature14236.

Narasimhan, K., Kulkarni, T. D., and Barzilay, R. Language understanding for text-based games using deep reinforcement learning. In Màrquez, L., Callison-Burch, C., Su, J., Pighin, D., and Marton, Y. (eds.), *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, EMNLP 2015, Lisbon, Portugal, September 17-21, 2015*, pp. 1–11. The Association for Computational Linguistics, 2015. doi: 10.18653/V1/D15-1001. URL https://doi.org/10.18653/v1/d15-1001.

Oh, J., Kim, S., and Yun, S. Fedbabu: Toward enhanced representation for federated image classification. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net, 2022. URL https://openreview.net/forum?id=HuaYQfggn5u.

Pan, J., Wang, X., Neubig, G., Jaitly, N., Ji, H., Suhr, A., and Zhang, Y. Training software engineering agents

and verifiers with swe-gym. In Singh, A., Fazel, M., Hsu, D., Lacoste-Julien, S., Berkenkamp, F., Maharaj, T., Wagstaff, K., and Zhu, J. (eds.), *Forty-second International Conference on Machine Learning, ICML 2025, Vancouver, BC, Canada, July 13-19, 2025*, volume 267 of *Proceedings of Machine Learning Research*. PMLR / OpenReview.net, 2025. URL https://proceedings.mlr.press/v267/pan25g.html.

Peng, H., Qi, Y., Wang, X., Yao, Z., Xu, B., Hou, L., and Li, J. Agentic reward modeling: Integrating human preferences with verifiable correctness signals for reliable reward systems. In Che, W., Nabende, J., Shutova, E., and Pilehvar, M. T. (eds.), *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2025, Vienna, Austria, July 27 - August 1, 2025*, pp. 15934–15949. Association for Computational Linguistics, 2025. URL https://aclanthology.org/2025.acl-long.775/.

Peng, J. et al. HiPER: Hierarchical reinforcement learning with explicit credit assignment for large language model agents. *arXiv preprint arXiv:2602.16165*, 2026a.

Peng, X., Chen, J., Wang, H., others, and Yao, H. SkillRL: Evolving agents via recursive skill-augmented reinforcement learning. *arXiv preprint arXiv:2602.08234*, 2026b.

Putta, P., Mills, E., Garg, N., Motwani, S., Finn, C., Garg, D., and Rafailov, R. Agent Q: advanced reasoning and learning for autonomous AI agents. *CoRR*, abs/2408.07199, 2024. doi: 10.48550/ARXIV.2408.07199. URL https://doi.org/10.48550/arXiv.2408.07199.

Qi, J., Zhou, Q., Lei, L., and Zheng, K. Federated reinforcement learning: Techniques, applications, and open challenges. *CoRR*, abs/2108.11887, 2021. URL https://arxiv.org/abs/2108.11887.

Qi, Z., Liu, X., Iong, I. L., Lai, H., Sun, X., Sun, J., Yang, X., Yang, Y., Yao, S., Xu, W., Tang, J., and Dong, Y. Webrl: Training LLM web agents via self-evolving online curriculum reinforcement learning. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net, 2025. URL https://openreview.net/forum?id=oVKEAFjEqv.

Qin, Z., Chen, D., Qian, B., Ding, B., Li, Y., and Deng, S. Federated full-parameter tuning of billion-sized language models with communication cost under 18 kilobytes. In Salakhutdinov, R., Kolter, Z., Heller, K. A., Weller, A., Oliver, N., Scarlett, J., and Berkenkamp, F. (eds.), *Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024*, volume 235 of *Proceedings of Machine Learning Research*, pp. 41473–41497. PMLR / OpenReview.net, 2024. URL https://proceedings.mlr.press/v235/qin24a.html.

Qin, Z., Wu, Z., He, B., and Deng, S. FedHDS: Federated data-efficient instruction tuning for large language models. In *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 15550–15568, 2025.

Rawles, C., Li, A., Rodriguez, D., Riva, O., and Lillicrap, T. P. Androidinthewild: A large-scale dataset for android device control. In Oh, A., Naumann, T., Globerson, A., Saenko, K., Hardt, M., and Levine, S. (eds.), *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, 2023. URL http://papers.nips.cc/paper_files/paper/2023/hash/bbbb6308b402fe909c39dd29950c32e0-Abstract-Datasets_and_Benchmarks.html.

Rothchild, D., Panda, A., Ullah, E., Ivkin, N., Stoica, I., Braverman, V., Gonzalez, J., and Arora, R. Fetchsgd: Communication-efficient federated learning with sketching. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, volume 119 of *Proceedings of Machine Learning Research*, pp. 8253–8265. PMLR, 2020. URL http://proceedings.mlr.press/v119/rothchild20a.html.

Sani, L., Iacob, A., Cao, Z., Lee, R., Marino, B., Gao, Y., Zhao, W., Cai, D., Li, Z., Qiu, X., and Lane, N. D. Photon: Federated LLM pre-training. In Zaharia, M., Joshi, G., and Lin, Y. C. (eds.), *Proceedings of the Eighth Conference on Machine Learning and Systems, MLSys 2025, Santa Clara, CA, USA, May 12-15, 2025*. OpenReview.net/mlsys.org, 2025. URL https://openreview.net/forum?id=AQgYcfg5EI.

Schick, T., Dwivedi-Yu, J., Dessì, R., Raileanu, R., Lomeli, M., Hambro, E., Zettlemoyer, L., Cancedda, N., and Scialom, T. Toolformer: Language models can teach themselves to use tools. In Oh, A., Naumann, T., Globerson, A., Saenko, K., Hardt, M., and Levine, S. (eds.), *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, 2023. URL http://papers.nips.cc/paper_files/paper/2023/hash/d842425e4bf79ba039352da0f658a906-Abstract-Conference.html.

Schulman, J., Wolski, F., Dhariwal, P., Radford, A., and Klimov, O. Proximal policy optimization algorithms. *CoRR*, abs/1707.06347, 2017. URL http://arxiv.org/abs/1707.06347.

Shao, Z., Wang, P., Zhu, Q., Xu, R., Song, J., Zhang, M., Li, Y. K., Wu, Y., and Guo, D. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *CoRR*, abs/2402.03300, 2024. doi: 10.48550/ARXIV.2402.03300. URL https://doi.org/10.48550/arXiv.2402.03300.

Shen, L., Tang, Z., Wu, L., Zhang, Y., Chu, X., Qin, T., and Han, B. Hot-pluggable federated learning: Bridging general and personalized FL via dynamic selection. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net, 2025a. URL https://openreview.net/forum?id=B8akWa62Da.

Shen, Z., Xu, T., Wang, H., Li, J., and Pan, M. pfedgpt: Hierarchically optimizing lora aggregation weights for personalized federated GPT models. In Christodoulopoulos, C., Chakraborty, T., Rose, C., and Peng, V. (eds.), *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing, EMNLP 2025, Suzhou, China, November 4-9, 2025*, pp. 4766–4778. Association for Computational Linguistics, 2025b. doi: 10.18653/V1/2025.EMNLP-MAIN.239. URL https://doi.org/10.18653/v1/2025.emnlp-main.239.

Shi, T., Chen, S., Jiang, B., Song, L., Yang, L., and Zhao, J. Experiential reinforcement learning. *arXiv preprint arXiv:2602.13949*, 2026.

Shi, Z., Wan, G., Huang, W., Zhang, G., Shao, J., Ye, M., and Yang, C. Privacy-enhancing paradigms within federated multi-agent systems. *CoRR*, abs/2503.08175, 2025. doi: 10.48550/ARXIV.2503.08175. URL https://doi.org/10.48550/arXiv.2503.08175.

Shinn, N., Cassano, F., Gopinath, A., Narasimhan, K., and Yao, S. Reflexion: language agents with verbal reinforcement learning. In Oh, A., Naumann, T., Globerson, A., Saenko, K., Hardt, M., and Levine, S. (eds.), *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, 2023. URL http://papers.nips.cc/paper_files/paper/2023/hash/1b44b878bb782e6954cd888628510e90-Abstract-Conference.html.

Shridhar, M., Yuan, X., Côté, M., Bisk, Y., Trischler, A., and Hausknecht, M. J. Alfworld: Aligning text and embodied environments for interactive learning. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net, 2021. URL https://openreview.net/forum?id=0IOX0YcCdTn.

Singh, A. et al. ARTIST: Agentic reasoning and tool integration for LLMs via reinforcement learning. *arXiv preprint arXiv:2505.01441*, 2025.

Singhal, R., Mohtashami, A., and Jaggi, M. FedEx-LoRA: Exact aggregation for federated and efficient fine-tuning of foundation models. In *Annual Meeting of the Association for Computational Linguistics*, 2025.

Song, Y., Yin, D., Yue, X., Huang, J., Li, S., and Lin, B. Y. Trial and error: Exploration-based trajectory optimization of LLM agents. In Ku, L., Martins, A., and Srikumar, V. (eds.), *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2024, Bangkok, Thailand, August 11-16, 2024*, pp. 7584–7600. Association for Computational Linguistics, 2024. doi: 10.18653/V1/2024.ACL-LONG.409. URL https://doi.org/10.18653/v1/2024.acl-long.409.

Spadea, F. and Seneviratne, O. Federated fine-tuning of large language models: Kahneman-tversky vs. direct preference optimization. In Long, G., Blumestein, M., Chang, Y., Lewin-Eytan, L., Huang, Z. H., and Yom-Tov, E. (eds.), *Companion Proceedings of the ACM on Web Conference 2025, WWW 2025, Sydney, NSW, Australia, 28 April 2025 - 2 May 2025*, pp. 1757–1760. ACM, 2025. doi: 10.1145/3701716.3717647. URL https://doi.org/10.1145/3701716.3717647.

Srewa, M., Zhao, T., and Elmalaki, S. Pluralllm: Pluralistic alignment in llms via federated learning. In *Proceedings of the 3rd International Workshop on Human-Centered Sensing, Modeling, and Intelligent Systems, HumanSys 2025, Irvine, CA, USA, May 6-9, 2025*, pp. 64–69. ACM, 2025. doi: 10.1145/3722570.3726898. URL https://doi.org/10.1145/3722570.3726898.

Stich, S. U. Local SGD converges fast and communicates little. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net, 2019. URL https://openreview.net/forum?id=S1g2JnRcFX.

Sun, Y., Li, Z., Li, Y., and Ding, B. Improving lora in privacy-preserving federated learning. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net, 2024. URL https://openreview.net/forum?id=NLPzL6HWNl.

Wan, Z., Li, Y., Song, Y., Wang, H., Yang, L., Schmidt, M., Wang, J., Zhang, W., Hu, S., and Wen, Y. Rema: Learning to meta-think for llms with multi-agent reinforcement learning. *CoRR*, abs/2503.09501, 2025. doi: 10.48550/ARXIV.2503.09501. URL https://doi.org/10.48550/arXiv.2503.09501.

Wang, H., Yurochkin, M., Sun, Y., Papailiopoulos, D. S., and Khazaeni, Y. Federated learning with matched averaging. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net, 2020a. URL https://openreview.net/forum?id=BkluqlSFDS.

Wang, H., He, S., Zhang, Z., Miao, F., and Anderson, J. Momentum for the win: Collaborative federated reinforcement learning across heterogeneous environments. In Salakhutdinov, R., Kolter, Z., Heller, K. A., Weller, A., Oliver, N., Scarlett, J., and Berkenkamp, F. (eds.), *Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024*, volume 235 of *Proceedings of Machine Learning Research*, pp. 50530–50560. PMLR / OpenReview.net, 2024a. URL https://proceedings.mlr.press/v235/wang24v.html.

Wang, J., Liu, Q., Liang, H., Joshi, G., and Poor, H. V. Tackling the objective inconsistency problem in heterogeneous federated optimization. In Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M., and Lin, H. (eds.), *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020b. URL https://proceedings.neurips.cc/paper/2020/hash/564127c03caab942e503ee6f810f54fd-Abstract.html.

Wang, K., Zhang, P., Wang, Z., Gao, Y., Li, L., Wang, Q., Chen, H., Wan, C., Lu, Y., Yang, Z., Wang, L., Krishna, R., Wu, J., Fei-Fei, L., Choi, Y., and Li, M. VAGEN: reinforcing world model reasoning for multi-turn VLM agents. *CoRR*, abs/2510.16907, 2025a. doi: 10.48550/ARXIV.2510.16907. URL https://doi.org/10.48550/arXiv.2510.16907.

Wang, L., Bian, J., Zhang, L., and Xu, J. FedLEASE: Adaptive LoRA experts allocation and selection for federated fine-tuning. In *Advances in Neural Information Processing Systems*, 2025b.

Wang, P., Li, L., Shao, Z., Xu, R., Dai, D., Li, Y., Chen, D., Wu, Y., and Sui, Z. Math-shepherd: Verify and reinforce llms step-by-step without human annotations. In Ku, L., Martins, A., and Srikumar, V. (eds.), *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2024, Bangkok, Thailand, August 11-16, 2024*, pp. 9426–9439. Association for Computational Linguistics, 2024b. doi: 10.18653/V1/2024.ACL-LONG.510. URL https://doi.org/10.18653/v1/2024.acl-long.510.

Wang, R. and Ammanabrolu, P. A practitioner's guide to multi-turn agentic reinforcement learning. *CoRR*, abs/2510.01132, 2025. doi: 10.48550/ARXIV.2510.01132. URL https://doi.org/10.48550/arXiv.2510.01132.

Wang, X., Wang, C., Li, X., Leung, V. C. M., and Taleb, T. Federated deep reinforcement learning for internet of things with decentralized cooperative edge caching. *IEEE Internet Things J.*, 7(10):9441–9455, 2020c. doi: 10.1109/JIOT.2020.2986803. URL https://doi.org/10.1109/JIOT.2020.2986803.

Wang, Z., Shen, Z., He, Y., Sun, G., Wang, H., Lyu, L., and Li, A. Flora: Federated fine-tuning large language models with heterogeneous low-rank adaptations. In Globersons, A., Mackey, L., Belgrave, D., Fan, A., Paquet, U., Tomczak, J. M., and Zhang, C. (eds.), *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*, 2024c. URL http://papers.nips.cc/paper_files/paper/2024/hash/28312c9491d60ed0c77f7fff4ad86dd1-Abstract-Conference.html.

Wang, Z., Wang, K., Wang, Q., Zhang, P., Li, L., Yang, Z., Jin, X., Yu, K., Nguyen, M. N., Liu, L., Gottlieb, E., Lu, Y., Cho, K., Wu, J., Fei-Fei, L., Wang, L., Choi, Y., and Li, M. RAGEN: understanding self-evolution in LLM agents via multi-turn reinforcement learning. *CoRR*, abs/2504.20073, 2025c. doi: 10.48550/ARXIV.2504.20073. URL https://doi.org/10.48550/arXiv.2504.20073.

Wang, Z., Xu, C., Liu, B., Wang, Y., Han, S., Yao, Z., Yao, H., and He, Y. Agent world model: Infinity synthetic environments for agentic reinforcement learning. *arXiv preprint arXiv:2602.10090*, 2026.

Wei, J., Wang, X., Schuurmans, D., Bosma, M., Ichter, B., Xia, F., Chi, E. H., Le, Q. V., and Zhou, D. Chain-of-thought prompting elicits reasoning in large language models. In Koyejo, S., Mohamed, S., Agarwal, A., Belgrave, D., Cho, K., and Oh, A. (eds.), *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*, 2022. URL http://papers.nips.cc/paper_files/paper/2022/hash/9d5609613524ecf4f15af0f7b31abca4-Abstract-Conference.html.

Wei, Q., Zeng, S., Li, C., Brown, W., Frunza, O., Deng, W., Nevmyvaka, Y., Zhao, Y., Garcia, A., and Hong, M. Reinforcing multi-turn reasoning in LLM agents via turn-level reward design. *arXiv preprint arXiv:2505.11821*, 2025a.

Wei, S., Tong, Y., Zhou, Z., Xu, Y., Gao, J., Wei, T., He, T., and Lv, W. Federated reasoning llms: a survey. *Frontiers Comput. Sci.*, 19(12):1912613, 2025b. doi: 10.1007/S11704-025-50480-3. URL https://doi.org/10.1007/s11704-025-50480-3.

Wei, Y., Duchenne, O., Copet, J., Carbonneaux, Q., Zhang, L., Fried, D., Synnaeve, G., Singh, R., and Wang, S. I. SWE-RL: advancing LLM reasoning via reinforcement learning on open software evolution. *CoRR*, abs/2502.18449, 2025c. doi: 10.48550/ARXIV.2502.18449. URL https://doi.org/10.48550/arXiv.2502.18449.

Wei, Y., Sun, Z., McMilin, E., Gehring, J., Zhang, D., Synnaeve, G., Fried, D., Zhang, L., and Wang, S. I. Toward training superintelligent software agents through self-play SWE-RL. *CoRR*, abs/2512.18552, 2025d. doi: 10.48550/ARXIV.2512.18552. URL https://doi.org/10.48550/arXiv.2512.18552.

Wei, Z., Yao, W., Liu, Y., Zhang, W., Lu, Q., Qiu, L., Yu, C., Xu, P., Zhang, C., Yin, B., Yun, H., and Li, L. Webagent-r1: Training web agents via end-to-end multi-turn reinforcement learning. In Christodoulopoulos, C., Chakraborty, T., Rose, C., and Peng, V. (eds.), *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing, EMNLP 2025, Suzhou, China, November 4-9, 2025*, pp. 7909–7928. Association for Computational Linguistics, 2025e. doi: 10.18653/V1/2025.EMNLP-MAIN.401. URL https://doi.org/10.18653/v1/2025.emnlp-main.401.

Wen, J., Dai, H., He, J., Xi, M., Xiao, S., and Yang, J. Federated offline reinforcement learning with multimodal data. *IEEE Trans. Consumer Electron.*, 70(1):4266–4276, 2024. doi: 10.1109/TCE.2023.3330943. URL https://doi.org/10.1109/TCE.2023.3330943.

Woo, J., Joshi, G., and Chi, Y. The blessing of heterogeneity in federated q-learning: Linear speedup and beyond. In Krause, A., Brunskill, E., Cho, K., Engelhardt, B., Sabato, S., and Scarlett, J. (eds.), *International Conference on Machine Learning, ICML 2023, 23-29 July 2023, Honolulu, Hawaii, USA*, volume 202 of *Proceedings of Machine Learning Research*, pp. 37157–37216. PMLR, 2023. URL https://proceedings.mlr.press/v202/woo23a.html.

Woodworth, B. E., Patel, K. K., and Srebro, N. Minibatch vs local SGD for heterogeneous distributed learning. In Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M., and Lin, H. (eds.), *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020. URL https://proceedings.neurips.cc/paper/2020/hash/45713f6ff2041d3fdfae927b82488db8-Abstract.html.

Wu, F., Li, Z., Li, Y., Ding, B., and Gao, J. Fedbiot: LLM local fine-tuning in federated learning without full model. In Baeza-Yates, R. and Bonchi, F. (eds.), *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, KDD 2024, Barcelona, Spain, August 25-29, 2024*, pp. 3345–3355. ACM, 2024a. doi: 10.1145/3637528.3671897. URL https://doi.org/10.1145/3637528.3671897.

Wu, F., Liu, X., Wang, H., Wang, X., Su, L., and Gao, J. Towards federated RLHF with aggregated client preference for llms. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net, 2025. URL https://openreview.net/forum?id=mqNKiEB6pd.

Wu, P., Li, K., Nan, J., and Wang, F. Federated in-context LLM agent learning. *CoRR*, abs/2412.08054, 2024b. doi: 10.48550/ARXIV.2412.08054. URL https://doi.org/10.48550/arXiv.2412.08054.

Wu, Q., Bansal, G., Zhang, J., Wu, Y., Zhang, S., Zhu, E., Li, B., Jiang, L., Zhang, X., and Wang, C. Autogen: Enabling next-gen LLM applications via multi-agent conversation framework. *CoRR*, abs/2308.08155, 2023. doi: 10.48550/ARXIV.2308.08155. URL https://doi.org/10.48550/arXiv.2308.08155.

Xi, Z., Ding, Y., Chen, W., Hong, B., Guo, H., Wang, J., Yang, D., Liao, C., Guo, X., He, W., Gao, S., Chen, L., Zheng, R., Zou, Y., Gui, T., Zhang, Q., Qiu, X., Huang, X., Wu, Z., and Jiang, Y. Agentgym: Evolving large language model-based agents across diverse environments. *CoRR*, abs/2406.04151, 2024. doi: 10.48550/ARXIV.2406.04151. URL https://doi.org/10.48550/arXiv.2406.04151.

Xi, Z., Huang, J., Liao, C., Huang, B., Guo, H., Liu, J., Zheng, R., Ye, J., Zhang, J., Chen, W., He, W., Ding, Y., Li, G., Chen, Z., Du, Z., Yao, X., Xu, Y., Chen, J., Gui, T., Wu, Z., Zhang, Q., Huang, X., and Jiang, Y. Agentgym-rl: Training LLM agents for long-horizon decision making through multi-turn reinforcement learning. *CoRR*, abs/2509.08755, 2025. doi: 10.48550/ARXIV.2509.08755. URL https://doi.org/10.48550/arXiv.2509.08755.

Xiao, Z., Tu, J., Zou, C., Zuo, Y., Li, Z., Wang, P., Yu, B., Huang, F., Lin, J., and Liu, Z. WebWorld: A large-scale world model for web agent training. *arXiv preprint arXiv:2602.14721*, 2026.

Xiong, B., Yang, X., Song, Y., Wang, Y., and Xu, C. Pilot: Building the federated multimodal instruction tuning framework. In Walsh, T., Shah, J., and Kolter, Z. (eds.), *AAAI-25, Sponsored by the Association for the Advancement of Artificial Intelligence, February 25 - March 4, 2025, Philadelphia, PA, USA*, pp. 21716–21724. AAAI Press, 2025a. doi: 10.1609/AAAI.V39I20.35476. URL https://doi.org/10.1609/aaai.v39i20.35476.

Xiong, G., Wang, S., Jiang, D., and Li, J. On the linear speedup of personalized federated reinforcement learning with shared representations. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net, 2025b. URL https://openreview.net/forum?id=BfUDZGqCAu.

Xu, F. F., Song, Y., Li, B., Tang, Y., Jain, K., Bao, M., Wang, Z. Z., Zhou, X., Guo, Z., Cao, M., Yang, M., Lu, H. Y., Martin, A., Su, Z., Maben, L., Mehta, R., Chi, W., Jang, L. K., Xie, Y., Zhou, S., and Neubig, G. Theagentcompany: Benchmarking LLM agents on consequential real world tasks. *CoRR*, abs/2412.14161, 2024. doi: 10.48550/ARXIV.2412.14161. URL https://doi.org/10.48550/arXiv.2412.14161.

Yan, Y., Feng, C., Zuo, W., Goh, R. S. M., Liu, Y., and Zhu, L. Federated residual low-rank adaptation of large language models. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net, 2025. URL https://openreview.net/forum?id=e0rQRMUhs7.

Yang, T., Cen, S., Wei, Y., Chen, Y., and Chi, Y. Federated natural policy gradient and actor critic methods for multi-task reinforcement learning. In Globersons, A., Mackey, L., Belgrave, D., Fan, A., Paquet, U., Tomczak, J. M., and Zhang, C. (eds.), *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*, 2024. URL http://papers.nips.cc/paper_files/paper/2024/hash/dbdea7859f1d2fc10f2c9e79b8f5ae54-Abstract-Conference.html.

Yang, Y., Long, G., Lu, Q., Zhu, L., Jiang, J., and Zhang, C. Federated low-rank adaptation for foundation models: A survey. In *Proceedings of the Thirty-Fourth International Joint Conference on Artificial Intelligence, IJCAI 2025, Montreal, Canada, August 16-22, 2025*, pp. 10779–10787. ijcai.org, 2025. doi: 10.24963/IJCAI.2025/1196. URL https://doi.org/10.24963/ijcai.2025/1196.

Yao, S., Chen, H., Yang, J., and Narasimhan, K. Webshop: Towards scalable real-world web interaction with grounded language agents. In Koyejo, S., Mohamed, S., Agarwal, A., Belgrave, D., Cho, K., and Oh, A. (eds.), *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*, 2022. URL http://papers.nips.cc/paper_files/paper/2022/hash/82ad13ec01f9fe44c01cb91814fd7b8c-Abstract-Conference.html.

Yao, S., Yu, D., Zhao, J., Shafran, I., Griffiths, T., Cao, Y., and Narasimhan, K. Tree of thoughts: Deliberate problem solving with large language models. In Oh, A., Naumann, T., Globerson, A., Saenko, K., Hardt, M., and Levine, S. (eds.), *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, 2023a. URL http://papers.nips.cc/paper_files/paper/2023/hash/271db9922b8d1f4dd7aaef84ed5ac703-Abstract-Conference.html.

Yao, S., Zhao, J., Yu, D., Du, N., Shafran, I., Narasimhan, K. R., and Cao, Y. React: Synergizing reasoning and acting in language models. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net, 2023b. URL https://openreview.net/forum?id=WE_vluYUL-X.

Ye, M., Fang, X., Du, B., Yuen, P. C., and Tao, D. Heterogeneous federated learning: State-of-the-art and research challenges. *ACM Comput. Surv.*, 56(3):79:1–79:44, 2024a. doi: 10.1145/3625558. URL https://doi.org/10.1145/3625558.

Ye, R., Wang, W., Chai, J., Li, D., Li, Z., Xu, Y., Du, Y., Wang, Y., and Chen, S. Openfedllm: Training large language models on decentralized private data via federated learning. In Baeza-Yates, R. and Bonchi, F. (eds.), *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, KDD 2024, Barcelona, Spain, August 25-29, 2024*, pp. 6137–6147. ACM, 2024b. doi: 10.1145/3637528.3671582. URL https://doi.org/10.1145/3637528.3671582.

Yi, X., Liu, Y., Yang, B., and Zhang, J. Federated continual learning via orchestrating multi-scale expertise. In *Advances in Neural Information Processing Systems*, 2025.

Yu, P., Wynter, L., and Lim, S. H. Federated reinforcement learning for portfolio management. In Ludwig, H. and Baracaldo, N. (eds.), *Federated Learning - A Comprehensive Overview of Methods and Applications*, pp. 467–482. Springer, 2022. doi: 10.1007/978-3-030-96896-0_21. URL https://doi.org/10.1007/978-3-030-96896-0_21.

Yu, Q., Zhang, Z., Zhu, R., Yuan, Y., Zuo, X., Yue, Y., Fan, T., Liu, G., Liu, L., Liu, X., Lin, H., Lin, Z., Ma, B., Sheng, G., Tong, Y., Zhang, C., Zhang, M., Zhang, W., Zhu, H., Zhu, J., Chen, J., Chen, J., Wang, C., Yu, H., Dai, W., Song, Y., Wei, X., Zhou, H., Liu, J., Ma, W., Zhang, Y., Yan, L., Qiao, M., Wu, Y., and Wang, M. DAPO: an open-source LLM reinforcement learning system at scale. *CoRR*, abs/2503.14476, 2025a. doi: 10.48550/ARXIV.2503.14476. URL https://doi.org/10.48550/arXiv.2503.14476.

Yu, X., Peng, B., Xu, R., Galley, M., Cheng, H., Nath, S., Gao, J., and Yu, Z. Dyna-think: Synergizing reasoning, acting, and world model simulation in AI agents. *CoRR*, abs/2506.00320, 2025b. doi: 10.48550/ARXIV.2506.00320. URL https://doi.org/10.48550/arXiv.2506.00320.

Yu, X., Peng, B., Xu, R., Shen, Y., He, P., Nath, S., Singh, N., Gao, J., and Yu, Z. Reinforcement world model learning for LLM-based agents. *arXiv preprint arXiv:2602.05842*, 2026.

Zeng, A., Liu, M., Lu, R., Wang, B., Liu, X., Dong, Y., and Tang, J. Agenttuning: Enabling generalized agent abilities for llms. In Ku, L., Martins, A., and Srikumar, V. (eds.), *Findings of the Association for Computational Linguistics, ACL 2024, Bangkok, Thailand and virtual meeting, August 11-16, 2024*, volume ACL 2024 of *Findings of ACL*, pp. 3053–3077. Association for Computational Linguistics, 2024. doi: 10.18653/V1/2024.FINDINGS-ACL.181. URL https://doi.org/10.18653/v1/2024.findings-acl.181.

Zhang, G., Geng, H., Yu, X., Yin, Z., Zhang, Z., Tan, Z., Zhou, H., Li, Z., Xue, X., Li, Y., Zhou, Y., Chen, Y., Zhang, C., Fan, Y., Wang, Z., Huang, S., Velez, F. P., Liao, Y., Wang, H., Yang, M., Ji, H., Wang, J., Yan, S., Torr, P., and Bai, L. The landscape of agentic reinforcement learning for llms: A survey. *Trans. Mach. Learn. Res.*, 2026, 2026a. URL https://openreview.net/forum?id=RY19y2RI1O.

Zhang, H., Liu, X., Lv, B., Sun, X., Jing, B., Iong, I. L., Hou, Z., Qi, Z., Lai, H., Xu, Y., Lu, R., Wang, H., Tang, J., and Dong, Y. Agentrl: Scaling agentic reinforcement learning with a multi-turn, multi-task framework. *CoRR*, abs/2510.04206, 2025a. doi: 10.48550/ARXIV.2510.04206. URL https://doi.org/10.48550/arXiv.2510.04206.

Zhang, H. et al. MARTI: A framework for multi-agent LLM systems reinforced training and inference. In *International Conference on Learning Representations*, 2026b.

Zhang, J., Vahidian, S., Kuo, M., Li, C., Zhang, R., Yu, T., Wang, G., and Chen, Y. Towards building the federatedgpt: Federated instruction tuning. In *IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2024, Seoul, Republic of Korea, April 14-19, 2024*, pp. 6915–6919. IEEE, 2024a. doi: 10.1109/ICASSP48485.2024.10447454. URL https://doi.org/10.1109/ICASSP48485.2024.10447454.

Zhang, K., Chen, X., Liu, B., Xue, T., Liao, Z., Liu, Z., Wang, X., Ning, Y., Chen, Z., Fu, X., Xie, J., Sun, Y., Gou, B., Qi, Q., Meng, Z., Yang, J., Zhang, N., Li, X., Shah, A., Huynh, D., Li, H., Yang, Z., Cao, S., Jang, L., Zhou, S., Zhu, J., Sun, H., Weston, J., Su, Y., and Wu, Y. Agent learning via early experience. *CoRR*, abs/2510.08558, 2025b. doi: 10.48550/ARXIV.2510.08558. URL https://doi.org/10.48550/arXiv.2510.08558.

Zhang, Y. et al. FedAMoLE: Personalized federated fine-tuning for LLMs via data-driven heterogeneous model architectures. *arXiv preprint arXiv:2411.19128*, 2024b.

Zhang, Z. et al. FewFedPIT: Towards privacy-preserving and few-shot federated instruction tuning. *arXiv preprint arXiv:2403.06131*, 2024c.

Zheng, Z., Gao, F., Xue, L., and Yang, J. Federated q-learning: Linear regret speedup with low communication cost. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net, 2024. URL https://openreview.net/forum?id=fe6ANBxcKM.

Zheng, Z., Zhang, H., and Xue, L. Federated q-learning with reference-advantage decomposition: Almost optimal regret and logarithmic communication cost. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net, 2025. URL https://openreview.net/forum?id=FoUpv84hMw.

Zhou, A., Yan, K., Shlapentokh-Rothman, M., Wang, H., and Wang, Y. Language agent tree search unifies reasoning, acting, and planning in language models. In Salakhutdinov, R., Kolter, Z., Heller, K. A., Weller, A., Oliver, N., Scarlett, J., and Berkenkamp, F. (eds.), *Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024*, volume 235 of *Proceedings of Machine Learning Research*, pp. 62138–62160. PMLR / OpenReview.net, 2024a. URL https://proceedings.mlr.press/v235/zhou24r.html.

Zhou, S., Xu, F. F., Zhu, H., Zhou, X., Lo, R., Sridhar, A., Cheng, X., Ou, T., Bisk, Y., Fried, D., Alon, U., and Neubig, G. Webarena: A realistic web environment for building autonomous agents. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net, 2024b. URL https://openreview.net/forum?id=oKn9c6ytLx.

Zhou, Y., Zanette, A., Pan, J., Levine, S., and Kumar, A. Archer: Training language model agents via hierarchical multi-turn RL. In Salakhutdinov, R., Kolter, Z., Heller, K. A., Weller, A., Oliver, N., Scarlett, J., and Berkenkamp, F. (eds.), *Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024*, volume 235 of *Proceedings of Machine Learning Research*, pp. 62178–62209. PMLR / OpenReview.net, 2024c. URL https://proceedings.mlr.press/v235/zhou24t.html.

Zhou, Y., Jiang, S., Tian, Y., Weston, J., Levine, S., Sukhbaatar, S., and Li, X. SWEET-RL: training multi-turn LLM agents on collaborative reasoning tasks. *CoRR*, abs/2503.15478, 2025. doi: 10.48550/ARXIV.2503.15478. URL https://doi.org/10.48550/arXiv.2503.15478.

Zhu, K., Du, H., Hong, Z., Yang, X., Guo, S., Wang, Z., Wang, Z., Qian, C., Tang, R., Ji, H., and You, J. Multiagentbench : Evaluating the collaboration and competition of LLM agents. In Che, W., Nabende, J., Shutova, E., and Pilehvar, M. T. (eds.), *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2025, Vienna, Austria, July 27 - August 1, 2025*, pp. 8580–8622. Association for Computational Linguistics, 2025. doi: 10.18653/V1/2025.ACL-LONG.421. URL https://doi.org/10.18653/v1/2025.acl-long.421.

Zhu, Y. and Gong, X. Single-loop federated actor-critic across heterogeneous environments. In Walsh, T., Shah, J., and Kolter, Z. (eds.), *AAAI-25, Sponsored by the Association for the Advancement of Artificial Intelligence, February 25 - March 4, 2025, Philadelphia, PA, USA*, pp. 23054–23062. AAAI Press, 2025. doi: 10.1609/AAAI.V39I21.34469. URL https://doi.org/10.1609/aaai.v39i21.34469.

Content of Appendix

A. Reproducibility Statement

We performed all experiments on NVIDIA H100 GPUs, using the HuggingFace Transformers framework (https://huggingface.co/docs/transformers/en/index) as the basis for our implementation. To support further verification, replication and development, we release all the dataset, code and experimental outputs at the repository: https://anonymous.4open.science/r/federated_agent_submission-4652/README.md.

B. Extended Related Work

B.1. Supervised Federated Learning

Federated learning (FL) was introduced by McMahan et al. (2017) with the Federated Averaging (FEDAVG) algorithm, which enables multiple clients to collaboratively train a shared model by communicating only model updates rather than raw data. In the canonical FL protocol, a central server coordinates N clients over T communication rounds: in each round, the server broadcasts the current global model $\boldsymbol{w}^t$ to a subset of selected clients; each selected client k performs E epochs of stochastic gradient descent on its local dataset $\mathcal{D}_k$ to obtain an updated model $\boldsymbol{w}_k^{t+1}$; and the server aggregates these local models via a weighted average $\boldsymbol{w}^{t+1} = \sum_k p_k \boldsymbol{w}_k^{t+1}$, where $p_k = |\mathcal{D}_k| / \sum_j |\mathcal{D}_j|$. This simple yet effective paradigm preserves data privacy by never transmitting raw data, and has since motivated a vast body of research addressing its key challenges: *statistical heterogeneity* arising from non-IID client data distributions, *systems heterogeneity* due to varying client computation and communication capabilities, and *privacy* concerns requiring formal guarantees beyond the implicit protection of keeping data local (Kairouz et al., 2021; Li et al., 2020a). The majority of FL research has focused on supervised classification tasks, establishing both theoretical foundations and practical algorithms.

Data Heterogeneity. A central challenge in supervised FL is *data heterogeneity*, where clients possess non-identically distributed (non-IID) data. This heterogeneity has been characterized along several axes, including label distribution skew (Hsu et al., 2019), feature distribution shift (Li et al., 2021c), and quantity imbalance (Gao et al., 2022; Ye et al., 2024a). Convergence analyses under non-IID data have been developed for FedAvg (Li et al., 2020c; Stich, 2019; Woodworth et al., 2020), establishing how heterogeneity slows convergence and motivating algorithm improvements. To address client drift, various methods have been proposed: FEDPROX (Li et al., 2020b) adds a proximal regularization term to the local objective; SCAFFOLD (Karimireddy et al., 2020) corrects for client drift using control variates; FEDNOVA (Wang et al., 2020b) normalizes local updates to handle heterogeneous local computation; FEDDYN (Acar et al., 2021) augments each device's objective with dynamic linear and quadratic penalty terms; and MOON (Li et al., 2021a) uses model-contrastive learning to align local and global representations.

Server-Side Optimization and Aggregation. Beyond local corrections, server-side optimization has also received attention. Chen et al. (2022) proposed FEDOPT, a general framework applying adaptive server-side optimizers (FedAdam, FedAdagrad, FedYogi) to federated learning. FEDMA (Wang et al., 2020a) aligns neural network layers via matched averaging, while knowledge distillation-based approaches such as FEDDF (Lin et al., 2020) aggregate client models via ensemble distillation on the server. FEDMD (Li & Wang, 2019) enables FL with heterogeneous model architectures by distilling averaged logits on a shared public dataset.

Personalization. Personalized federated learning methods further address heterogeneity by learning client-specific models while still benefiting from collaborative training. Per-FedAvg (Fallah et al., 2020) formulates personalized FL as a MAML-style meta-learning problem; pFedMe (Dinh et al., 2020) decouples personalized model optimization from global model learning using Moreau envelopes; FedRep (Collins et al., 2021) and FedBABU (Oh et al., 2022) learn a shared representation (body) while maintaining client-specific heads; and Ditto (Li et al., 2021b) balances global and local objectives for fair and robust personalization. Other approaches include FedPer (Arivazhagan et al., 2019) with personalization layers and APFL (Deng et al., 2020) with adaptive mixture weights. More recently, Hot-Pluggable FL (Shen et al., 2025a) proposes a plug-in module framework that bridges general and personalized FL via dynamic module selection from a shared modular store, and FOL (Huang & Shu, 2025) introduces a practical one-shot personalized FL framework achieving strong personalization with only a single round of model exchange. Clustered FL has also advanced with HCFL (Guo et al., 2025b), which proposes a four-tier framework that systematically integrates and extends clustering strategies. MultiFCL (Yi et al., 2025) addresses federated continual learning by orchestrating multi-scale expertise with lightweight frozen adapters to prevent catastrophic forgetting across sequential tasks.

Communication Efficiency and Privacy. Reducing communication overhead has also been extensively studied. Gradient compression (Konečný et al., 2016; Lin et al., 2018), quantization (Alistarh et al., 2017), and sketching (Rothchild et al., 2020) can reduce communication cost by orders of magnitude. Privacy-preserving techniques such as differential privacy (McMahan et al., 2018; Geyer et al., 2017) have been integrated into FL to provide formal privacy guarantees.

Positioning of FEDAGENT. Despite this extensive body of work, all supervised FL methods are designed for *static, one-shot prediction* tasks with well-defined input–label pairs. In contrast, FEDAGENT operates in *sequential decision-making*

settings where agents interact with environments over multiple steps, producing natural language actions and receiving sparse rewards. The heterogeneity challenges we define, Preference, Coverage, and Hardness Heterogeneity, are fundamentally distinct from label/feature/quantity skew and require new partition strategies for systematic study.

B.2. Federated Reinforcement Learning

Federated reinforcement learning (FRL) extends federated learning to sequential decision-making problems, where multiple agents learn a shared policy by training in their own environments without sharing raw trajectories (Qi et al., 2021). In contrast to supervised FL, which operates on static datasets with well-defined input–label pairs, FRL must contend with the unique challenges of RL: agents interact with potentially different Markov Decision Processes (MDPs) $\mathcal{M}_k = (\mathcal{S}_k, \mathcal{A}_k, P_k, R_k, \gamma)$, generating non-stationary trajectory data that depends on the evolving policy. This introduces several distinctive difficulties beyond standard FL, including (i) *non-stationarity* of the data distribution as the policy changes during training, (ii) *environment heterogeneity* where clients may have different transition dynamics P_k or reward functions R_k, (iii) the need for *exploration-exploitation tradeoffs* that must be coordinated across agents, and (iv) *credit assignment* over temporally extended trajectories. These challenges have motivated a growing body of theoretical and algorithmic work (Liu et al., 2025a; Gao et al., 2025).

Foundations and Theory. Early work by Yu et al. (2022) proposed sharing experiences through an encrypted channel for collaborative multi-task RL. Jin et al. (2022) studied federated RL in linear MDPs, providing sample complexity analysis. Khodadadian et al. (2022) analyzed the convergence of federated temporal-difference learning with linear speedup under Markovian sampling. Woo et al. (2023) further studied the sample complexity of heterogeneous FRL and showed that heterogeneity can in fact help exploration. More recently, Zheng et al. (2024) proposed FedQ-Hoeffding and FedQ-Bernstein algorithms for tabular episodic MDPs that achieve linear regret speedup with logarithmic communication cost, and Zheng et al. (2025) further achieved almost minimax-optimal regret via reference-advantage decomposition. Labbi et al. (2025) extended UCBVI to the federated setting with communication-efficient regret minimization.

Federated Policy Gradient Methods. Several works have developed federated policy optimization algorithms. Yang et al. (2024) developed federated natural policy gradient (NPG) and natural actor-critic (NAC) methods with the first global convergence guarantees for federated multi-task RL. Lan et al. (2025) proposed AFedPG, an asynchronous FRL framework that achieves linear speedup under heterogeneous computing. Zhu & Gong (2025) introduced a single-loop federated actor-critic method that jointly considers both federated policy evaluation and improvement. Fan et al. (2021) proposed the first Byzantine fault-tolerant FRL framework, proving linear speedup even when less than half of agents are adversarial.

Heterogeneous Environments. Handling environment heterogeneity across clients is a key challenge. Wang et al. (2024a) proposed momentum-based federated policy gradient methods that converge to a stationary point regardless of the magnitude of environment heterogeneity. Xiong et al. (2025b) introduced personalized FRL with shared representations, proving the first linear speedup result in the personalized FRL setting. Liu et al. (2019) explored lifelong federated RL for cloud robotic systems, where heterogeneous robots share knowledge to accelerate learning.

Applications. FRL has been applied to diverse domains including autonomous driving (Liang et al., 2019), network resource management (Wang et al., 2020c), recommendation systems (Li et al., 2024), and federated offline RL for healthcare (Wen et al., 2024). Hwang & Hong (2025) proposed FedRQ for tabular FRL with proven asymptotic convergence under environment heterogeneity, extended to continuous state spaces via expectile loss. Recent work has also explored offline federated deep RL (Chen et al., 2025e), which aggregates multiple client-side offline RL models by considering both Q-values and policy inconsistency when assigning client aggregation weights. A comprehensive survey by Cheruiyot et al. (2025) provides a unified perspective on the connections between federated RL, cooperative decentralized RL, and noncooperative multi-agent RL paradigms.

Positioning of FEDAGENT. Despite this growing body of work, existing FRL methods are designed for environments with *low-dimensional, discrete or continuous state-action spaces* and *dense reward signals*. In contrast, FEDAGENT addresses LLM agent training, where (i) actions are *long sequences of natural language tokens*, making the action space exponentially large; (ii) rewards are typically *sparse and episode-level*; and (iii) credit assignment must operate at the *token level* across multi-step reasoning traces. These fundamental differences make prior FRL approaches not directly applicable to the agent learning setting that FEDAGENT addresses.

B.3. Federated Learning for LLMs

The intersection of federated learning and large language models (LLMs) has recently attracted growing attention (Zhang et al., 2024a; Kuang et al., 2024; Ye et al., 2024b). As LLMs become increasingly central to real-world applications, the need to fine-tune them on distributed private data, such as enterprise documents, clinical records, and user interaction logs, makes federated learning a natural paradigm. However, applying FL to LLMs introduces unique challenges beyond classical FL settings: (i) the *massive parameter count* of modern LLMs (7B–70B+ parameters) makes communicating full model updates infeasible, necessitating parameter-efficient methods; (ii) *heterogeneous client capabilities* mean that some clients can only fine-tune small adapters while others may handle larger updates; (iii) LLM training objectives span *supervised instruction tuning, preference optimization, and reinforcement learning from human feedback (RLHF)*, each requiring different federated protocols; and (iv) *privacy requirements* are often stricter for LLMs due to the sensitivity of natural language data and the risk of memorization. These challenges have spurred rapid development of federated LLM training methods across multiple research communities (Wei et al., 2025b).

Federated Instruction Tuning. A natural first step is to apply FL to supervised instruction tuning. Zhang et al. (2024a) proposed FedIT for federated instruction tuning with LoRA adapters on heterogeneous instruction datasets. Ye et al. (2024b) introduced OpenFedLLM, a training pipeline supporting federated SFT and instruction tuning across diverse datasets. A key challenge in federated instruction tuning is *data quality and selection*: FedHDS (Qin et al., 2025) selects representative subsets at both intra- and inter-client levels without sharing raw data, achieving 10.72% improvement in Rouge-L on unseen tasks while using less than 1.5% of the data; FedDQC (Du et al., 2025) introduces an instruction-response alignment metric for efficient client-side quality evaluation and designs a quality-aware hierarchical training framework that progressively fine-tunes from high- to low-quality data. To address *data scarcity*, FewFedPIT (Zhang et al., 2024c) uses LLMs' in-context learning capability to self-generate synthetic data, combined with parameter-isolated training that separates public parameters from private ones to thwart data extraction attacks. On the *communication efficiency* front beyond LoRA-based methods, FedKSeed (Qin et al., 2024) employs zeroth-order optimization with a finite set of random seeds, reducing communication to under 18 kilobytes per round while enabling full-parameter tuning of billion-sized LLMs; FedMeZO (Ling et al., 2024) provides the first convergence analysis of memory-efficient zeroth-order optimization in federated LLM tuning, demonstrating faster convergence than first-order FedAvg with remarkably reduced GPU memory requirements. FedAMoLE (Zhang et al., 2024b) proposes a personalized FL framework featuring a heterogeneous mixture of LoRA experts module with a reverse selection-based expert assignment strategy that allows domain experts to select aligned clients. Pilot (Xiong et al., 2025a) extends federated instruction tuning to the multimodal setting, building a comprehensive framework for federated vision-language model fine-tuning that handles heterogeneous multimodal data across clients.

Parameter-Efficient Federated Fine-Tuning. Since communicating full LLM parameters is prohibitively expensive, parameter-efficient methods have become central. Sun et al. (2024) proposed FFA-LoRA, which freezes the randomly initialized matrix A and only fine-tunes the zero-initialized matrix B, halving communication and computation costs while working well under differential privacy. Wang et al. (2024c) introduced FLoRA, a stacking-based aggregation method that achieves noise-free aggregation and supports heterogeneous LoRA ranks. Bai et al. (2024b) proposed FlexLoRA, which dynamically adjusts local LoRA ranks to harness diverse client resources via SVD-based weight redistribution. Guo et al. (2025a) proposed FedSA-LoRA, which shares only the A matrices for server aggregation while keeping B matrices local, and Yan et al. (2025) introduced FRLoRA, performing global updates in a higher-rank parameter space via residual low-rank adaptation. Cho et al. (2024) proposed HetLoRA, allowing heterogeneous LoRA ranks across clients matching individual system resources. Hyeon-Woo et al. (2022) re-parameterizes weight layers using low-rank weights followed by the Hadamard product, achieving 3–10$\times$ lower communication cost. More recently, FedEx-LoRA (Singhal et al., 2025) achieves exact LoRA aggregation by adding a residual error term to frozen weights, eliminating the inexactness of traditional federated averaging of LoRA adapters. FedALT (Bian et al., 2025) departs from FedAvg by maintaining individual LoRA adapters alongside a shared Rest-of-World LoRA component with an adaptive MoE-style mixer. FedLEASE (Wang et al., 2025b) adaptively clusters clients based on representation similarity to allocate domain-specific LoRA experts with a top-M MoE mechanism. pFedGPT (Shen et al., 2025b) hierarchically optimizes LoRA aggregation weights for personalized federated GPT models. EcoLoRA (Liu et al., 2025b) uses round-robin segment sharing where each client uploads only a complementary LoRA segment per round, reducing communication time by up to 79%. FedICU (Liao et al., 2025) introduces importance-aware splitting and updating to dynamically balance LoRA components based on their contribution to global model performance, preventing catastrophic forgetting and domain overfitting. A comprehensive survey on federated LoRA methods was provided by Yang et al. (2025).

Federated Preference Optimization and Alignment. For alignment, FedRLHF (Fan et al., 2025) introduced a federated framework for privacy-preserving RLHF, demonstrating convergence guarantees and personalization capabilities across clients; however, it focuses on *single-turn preference alignment* on small-scale tasks (e.g., movie recommendations) rather than multi-step agent training. Wu et al. (2025) proposed FedBiscuit, which encodes each client's preferences into binary selectors and aggregates them to capture common preferences, establishing the first federated RLHF benchmark with heterogeneous human preference data. Spadea & Seneviratne (2025) evaluated Kahneman-Tversky Optimization (KTO) against DPO in federated settings, finding KTO more robust to data heterogeneity. Srewa et al. (2025) proposed PluralLLM, using federated learning for pluralistic LLM alignment that preserves group fairness.

Federated LLM Pre-Training and Knowledge Distillation. Sani et al. (2025) presented Photon, the first complete system for federated end-to-end LLM pre-training from scratch, training models up to 7B parameters. Wu et al. (2024a) proposed FedBiOT, where the server generates a compressed LLM via knowledge distillation and distributes it to clients who fine-tune only a lightweight adapter. A comprehensive survey by Wei et al. (2025b) organizes federated learning for reasoning LLMs by training signal types, covering federated pre-training, instruction tuning, prompt learning, adapter learning, and value alignment.

Federated Agent Systems. Very recently, a few works have begun to explore the intersection of federated learning and LLM agents. Fed-SE (Chen et al., 2025c) proposes a federated self-evolution framework where agents perform parameter-efficient fine-tuning on filtered high-return trajectories locally and aggregate globally, improving average task success rates by $\sim$18% over FedAvg across five heterogeneous environments. FICAL (Wu et al., 2024b) takes a different approach by transmitting knowledge compendiums rather than model parameters via in-context learning, combined with RAG-based tool learning. EPEAgent (Shi et al., 2025) introduces embedded privacy-enhancing agents that integrate into the RAG phase of federated multi-agent systems, minimizing data flows by ensuring only task-relevant, agent-specific information is shared.

Positioning of FEDAGENT. Despite this extensive progress, the existing works on federated LLM training remain largely limited to *supervised fine-tuning*, *single-turn RLHF*, or *preference optimization*. While a few concurrent works (Chen et al., 2025c; Wu et al., 2024b; Shi et al., 2025) have begun exploring federated agent learning, they either focus on fine-tuning via filtered trajectories without systematic RL training, rely on in-context learning rather than policy gradient optimization, or address privacy in retrieval rather than training. None of them provide a *comprehensive benchmark* with systematic heterogeneity definitions tailored to agent learning. FEDAGENT fills this gap by formulating a complete federated agent RL framework with a systematic benchmark (FEDAGENTGYM) featuring three novel heterogeneity challenges (Preference, Coverage, and Hardness Heterogeneity) that are unique to the agent learning setting.

B.4. LLM Agent Reinforcement Learning

LLM-based agents have demonstrated impressive capabilities in complex, interactive tasks that require multi-step reasoning, tool use, and environment interaction (Xi et al., 2024). Unlike standard LLM applications that generate single-turn responses, LLM agents operate in a *closed-loop* fashion: they observe an environment state, generate an action (typically a natural language command or API call), receive feedback from the environment, and iterate over multiple turns until a task is completed or a horizon limit is reached. This agentic paradigm has been applied to diverse domains including web navigation (Zhou et al., 2024b; Qi et al., 2025), software engineering (Jimenez et al., 2024; Wei et al., 2025c), embodied household tasks (Shridhar et al., 2021), mobile device control (Rawles et al., 2023; Bai et al., 2024a), and scientific research. Training LLM agents presents unique challenges compared to standard LLM fine-tuning: (i) the *action space is exponentially large*, as each action is a variable-length sequence of natural language tokens; (ii) rewards are typically *sparse and delayed*, provided only at the end of multi-step episodes; (iii) *credit assignment* must operate across both the token level within a single action and the turn level across the full interaction trajectory; and (iv) the agent must learn to balance *exploration* of new strategies with *exploitation* of known successful patterns. Two main paradigms have emerged for agent training: supervised fine-tuning on expert demonstrations and reinforcement learning from environment interaction.

Agent Frameworks. Prompting-based methods such as ReAct (Yao et al., 2023b), Reflexion (Shinn et al., 2023), Chain-of-Thought (Wei et al., 2022), and Tree of Thoughts (Yao et al., 2023a) improve agent reasoning without parameter updates, but their performance is bounded by the underlying model's capabilities. Schick et al. (2023) demonstrated that LLMs can learn to use external tool APIs in a self-supervised manner. Zhou et al. (2024a) unified reasoning, acting, and planning via Monte Carlo Tree Search with LM-powered value functions. Multi-agent systems such as AutoGen (Wu et al., 2023) enable

complex workflows through conversable agents.

Agent Training via Supervised Fine-Tuning. SFT methods leverage expert demonstrations or successful trajectories to train agents via imitation learning (Zeng et al., 2024; Chen et al., 2023). Song et al. (2024) proposed ETO, which learns from contrastive pairs of successful expert and failed agent trajectories using DPO. Zhang et al. (2025b) proposes "early experience" as a middle ground between expert demonstrations and full RL, using interaction data generated by the agent's own actions where resulting future states serve as supervision without reward signals. However, SFT-based methods are fundamentally limited by the quality and coverage of demonstrations and cannot improve beyond the expert's behavior.

RL Policy Optimization for LLM Agents. RL offers a principled framework for agents to improve through trial-and-error interaction with environments. Traditional RL methods such as PPO (Schulman et al., 2017) have been adapted for LLM agent training, but they face challenges with the vast token-level action space and sparse rewards in multi-step tasks. The success of DeepSeek-R1 (DeepSeek-AI et al., 2025) has demonstrated the power of RL in incentivizing complex reasoning in LLMs, spurring rapid follow-up work. GRPO (Shao et al., 2024) eliminates the need for a separate critic model by using group-relative advantages, and has become one of the most widely adopted algorithms for agent RL. DAPO (Yu et al., 2025a) introduces clip-higher and dynamic sampling techniques that achieve strong reasoning at scale. GiGPO (Feng et al., 2025) further improves upon GRPO with group-in-group advantage estimation for finer-grained credit assignment. RLOO (Ahmadian et al., 2024) proposes a parameter-free REINFORCE Leave-One-Out baseline that outperforms PPO while using substantially less GPU memory. Zhou et al. (2024c) proposed ArCHer, a hierarchical multi-turn RL framework, and Putta et al. (2024) introduced Agent Q with advanced reasoning for autonomous agents. Wang et al. (2025c) introduced StarPO, a general trajectory-level agent RL framework, identifying the "Echo Trap" failure mode. Agent-R1 (Cheng et al., 2025) extends single-turn RL to multi-turn agent settings by formalizing the LLM agent MDP.

Reward Design and Credit Assignment. The design of reward signals and credit assignment strategies is crucial for effective agent RL, given the sparse and delayed nature of episode-level feedback. Process reward models (Lightman et al., 2024; Wang et al., 2024b) provide step-level supervision that significantly outperforms outcome-only rewards. iStar (Liu et al., 2025d) proposes implicit step rewards that jointly optimize an implicit process reward model with the policy model via multi-turn DPO, achieving state-of-the-art on WebShop and SOTOPIA. Agentic Reward Modeling (Peng et al., 2025) combines human preference rewards with verifiable correctness signals from factuality and instruction following. For multi-turn credit assignment, Wei et al. (2025a) provides the first systematic study of turn-level reward design, extending GRPO and PPO to multi-turn variants. Wang & Ammanabrolu (2025) offers a practitioner's guide that systematically analyzes the design space across environment, reward, and policy dimensions. SWEET-RL (Zhou et al., 2025) uses a critic model with access to training-time information to provide step-level rewards for collaborative reasoning tasks. HiPER (Peng et al., 2026a) introduces hierarchical RL that separates high-level planning from low-level execution, achieving 97.4% on ALFWorld and 83.3% on WebShop.

World Models and Experience Efficiency. A key bottleneck of model-free agent RL is the reliance on expensive real-environment rollouts. An emerging line of work addresses this by integrating *world models* into the agent RL loop or improving *experience efficiency*. SPA (Chen et al., 2025b) cold-starts the policy via a self-play SFT stage to internalize a world model, then uses it to simulate future states prior to policy optimization. Dyna-Think (Yu et al., 2025b) integrates planning with an internal world model into the reasoning-acting loop, using Dyna-GRPO for online RL and achieving comparable best-of-n performance to R1-style reasoning with $2\times$ fewer tokens. DynaWeb (Ding et al., 2026) applies the Dyna architecture to web agents, training a dedicated world model to generate multi-step imagined trajectories. WebWorld (Xiao et al., 2026) trains the first open-web simulator at scale (up to 32B parameters on 1M+ real-world trajectories), supporting 30+ step long-horizon simulations; a 14B agent trained on WebWorld-synthesized trajectories improves by 9.2% on WebArena, matching GPT-4o. Agent World Model (Wang et al., 2026) proposes a fully synthetic environment generation pipeline scaling to 1,000 code-driven environments, demonstrating that training exclusively in synthetic environments yields strong out-of-distribution generalization. RWML (Yu et al., 2026) proposes self-supervised action-conditioned world model learning using sim-to-real gap rewards, outperforming direct RL on ALFWorld and τ-Bench. On the experience efficiency side, DreamGym (Chen et al., 2025d) introduces the first unified framework for synthesizing diverse agent experiences at scale, distilling environment dynamics into a reasoning-based experience model, outperforming all baselines by over 30% on non-RL-ready tasks. ERL (Shi et al., 2026) augments trial-and-error RL with an explicit experience-reflection-consolidation loop, achieving gains of up to 81% in complex multi-step environments. SkillRL (Peng et al., 2026b) bridges raw experience and policy improvement through automatic skill discovery and a hierarchical skill

library that co-evolves with the policy during RL.

Agent RL for Web, Code, and Reasoning. Agent RL has been successfully applied across diverse interactive domains. For *web agents*, WebRL (Qi et al., 2025) introduces a self-evolving online curriculum RL framework with an outcome-supervised reward model, improving Llama-3.1-8B from 4.8% to 42.4% success rate on WebArena-Lite. WebAgent-R1 (Wei et al., 2025e) demonstrates that simple end-to-end multi-turn RL with binary success rewards can boost web agent performance from 8.5% to 44.8%. For *device-control agents*, DigiRL (Bai et al., 2024a) proposes a two-stage offline-to-online RL approach. For *software engineering*, SWE-RL (Wei et al., 2025c) scales RL-based LLM reasoning for real-world software tasks, and Self-Play SWE-RL (Wei et al., 2025d) further extends it with a self-play paradigm for autonomous bug injection and repair without human labels. SWE-Gym (Pan et al., 2025) provides the first training environment for real-world software engineering agents with 2,438 executable task instances. DeepSWE (Luo et al., 2025) trains a fully open-source RL-based coding agent achieving 59% on SWE-Bench-Verified. Golubev et al. (2025) combines rejection fine-tuning with DAPO to train a 72B SWE agent, increasing SWE-bench pass rate from 20% to 39%. For *reasoning and tool use*, Search-R1 (Jin et al., 2025) extends R1-style RL to search-augmented settings, improving performance by 41% over RAG baselines. ARTIST (Singh et al., 2025) unifies agentic reasoning and tool integration via RL, enabling autonomous tool invocation within multi-turn reasoning chains. ML-Agent (Liu et al., 2025e) applies RL to autonomous machine learning engineering, where a 7B agent trained on merely 9 ML tasks outperforms the 671B DeepSeek-R1 agent. AgentRefine (Fu et al., 2025) trains agents to self-refine erroneous actions based on environment feedback.

Scaling and Training Infrastructure. Scaling agent RL to multi-turn, multi-task, and long-horizon settings introduces significant engineering and algorithmic challenges. AgentRL (Zhang et al., 2025a) presents a scalable multi-turn, multi-task framework with a fully-asynchronous generation-training pipeline, cross-policy sampling for exploration, and task advantage normalization for stable multi-task training. LOOP (Chen et al., 2025a) proposes a data- and memory-efficient PPO variant for interactive digital agents, with a 32B agent outperforming OpenAI o1 on AppWorld. AgentGym-RL (Xi et al., 2025) proposes ScalingInter-RL that gradually shifts from exploitation to exploration across training, matching or surpassing commercial models on 27 tasks. ReMA (Wan et al., 2025) decouples reasoning into a meta-thinking agent and a low-level reasoning agent via multi-agent RL. MAGRPO (Liu et al., 2025c) models LLM collaboration as cooperative multi-agent RL with centralized group-relative advantages. MARTI (Zhang et al., 2026b) provides an open-source framework for training multi-agent LLM systems with graph-based workflows and both rule-based and generative rewards. VAGEN (Wang et al., 2025a) introduces world modeling RL for multi-turn VLM agents.

Benchmarks and Evaluation. Several benchmarks have been established to evaluate agent capabilities, including WebShop (Yao et al., 2022) for web-based shopping tasks, ALFWorld (Shridhar et al., 2021) for embodied household tasks, AndroidInTheWild (Rawles et al., 2023) for mobile device control, SWE-bench (Jimenez et al., 2024) for real-world software engineering, WebArena (Zhou et al., 2024b) for realistic web navigation, AgentBench (Liu et al., 2024b) for multi-dimensional agent evaluation, and GAIA (Mialon et al., 2024) for general AI assistants. More recently, TheAgentCompany (Xu et al., 2024) simulates a full software company environment with integrated development and communication tools, where the best agents solve only 30% of tasks autonomously. AgentGym (Xi et al., 2024) provides a unified framework featuring 7 scenarios, 14 environments, and 89 tasks for concurrent agent interaction and self-evolution. MultiAgentBench (Zhu et al., 2025) evaluates multi-agent LLM systems across diverse collaboration and competition scenarios with novel milestone-based KPIs.

Positioning of FEDAGENT. Despite the rapid progress in LLM agent RL, from policy optimization algorithms (Shao et al., 2024; Yu et al., 2025a) and reward design (Wei et al., 2025a; Liu et al., 2025d) to world model augmentation (Chen et al., 2025b; Xiao et al., 2026) and scalable training frameworks (Zhang et al., 2025a; Xi et al., 2025), all existing methods assume a *centralized training* paradigm where all training data (task queries and trajectories) are collected in one place. This assumption is increasingly unrealistic due to privacy regulations (e.g., GDPR, CCPA) and the sensitivity of user interaction data. FEDAGENT addresses this critical limitation by enabling collaborative agent RL training across distributed clients while keeping all data local, providing the first systematic investigation of decentralized agent learning with a comprehensive benchmark featuring novel heterogeneity challenges unique to the agent setting.

C. More Details of Heterogeneity Challenges

C.1. Pseudo Code for Client Partitioning Strategies

C.1.1. PREFERENCEPARTITION

Algorithm 2 PREFERENCEPARTITION

Require: Category pools $\{\mathcal{I}_c\}_{c=1}^{C}$ with sizes n_c; total clients K; per-client set size L; jitter ω
Ensure: Client datasets $X_1, \ldots, X_K$ with $|X_k| = L$

$p_c \leftarrow n_c / \sum_{j=1}^{C} n_j$; $\quad \ell_c \leftarrow \log \frac{p_c}{1-p_c}$ ▷ global mix + logit anchors
for $k = 1$ to K **do**
  $z_c \sim \mathcal{N}(\ell_c, \omega^2)$ for $c = 1 \ldots C$; $\quad q_c \leftarrow \exp(z_c) / \sum_j \exp(z_j)$ ▷ larger $\omega \Rightarrow$ higher variance
  $(a_1, \ldots, a_C) \sim \mathrm{Multinomial}(L;\, q_1, \ldots, q_C)$ ▷ category counts for client k
  if any $a_c > n_c$ **then** set $a_c \leftarrow \min(a_c, n_c)$ and redistribute leftover by q to classes with spare capacity ▷ capacity fix within a set
  $X_k \leftarrow \bigcup_{c=1}^{C}$ SAMPLEWITHOUTREPLACEMENT$(\mathcal{I}_c, a_c)$
end for
return $\{X_k\}_{k=1}^{K}$

C.1.2. COVERAGEPARTITION

Algorithm 3 COVERAGEPARTITION

Require: total items N (indexed $1{:}N$); total clients K; per-client bounds $(L_{\min}, L_{\text{avg}}, L_{\max})$ with $L_{\min} \le L_{\text{avg}} \le L_{\max}$; dispersion ξ; desired average replicas per item r
Ensure: Client datasets $X_1, \ldots, X_K$

$T \leftarrow \lfloor rN \rceil$ ▷ total assignments (sum of all $|X_k|$); keeps global overlap fixed
assert $KL_{\min} \le T \le KL_{\max}$ ▷ feasibility under per-client bounds
$\mu \leftarrow (L_{\text{avg}} - L_{\min})/(L_{\max} - L_{\min})$; $\quad \alpha \leftarrow \mu\xi,\ \beta \leftarrow (1-\mu)\xi$ ▷ Beta params with mean fixed at L_{avg}
Sample $x_k \sim \mathrm{Beta}(\alpha, \beta)$ for $k = 1 \ldots K$ ▷ larger $\xi \Rightarrow$ lower variance (sizes closer to L_{avg})
$u_k \leftarrow L_{\min} + x_k(L_{\max} - L_{\min})$; $\quad u_k \leftarrow u_k \cdot \dfrac{T}{\sum_j u_j}$ ▷ shape then renormalize to sum T
$n_k \leftarrow$ ROUNDTOSUM$(u,\, T,\, [L_{\min}, L_{\max}])$ ▷ largest remainder with clipping to $[L_{\min}, L_{\max}]$
$m \leftarrow \lfloor r \rfloor$, $\quad M \leftarrow \lceil r \rceil$, $\quad H \leftarrow T - mN$
Set $q_i \leftarrow M$ for any H items; $q_i \leftarrow m$ otherwise
Initialize $X_k \leftarrow \emptyset$, $\mathrm{rem}_k \leftarrow n_k$ for all k
for $i = 1$ to N **do** ▷ weighted, no-replacement placement across clients
  $\mathcal{A} \leftarrow \{k : \mathrm{rem}_k > 0\}$; choose q_i distinct $k \in \mathcal{A}$ with $\Pr(k) \propto \mathrm{rem}_k$
  Add item i to each chosen X_k and decrement the corresponding rem_k
end for
return $\{X_k\}_{k=1}^{K}$

C.1.3. HARDNESSPARTITION

Algorithm 4 HARDNESSPARTITION

Require: total items N (indexed $1{:}N$); disjoint index sets $\mathcal{S}$ (successful) and $\mathcal{U}$ (unsuccessful) with $\mathcal{S} \cup \mathcal{U} = \{1{:}N\}$; total clients K; per-client set size L; Hyperparameters for COVERAGEPARTITION: bounds (ℓ, c, h) with $h \leq L$, dispersion ξ', overlap r

Ensure: client datasets $X_1, \ldots, X_K$ with $|X_k| = L$

$\{Y_k\}_{k=1}^K \leftarrow$ COVERAGEPARTITION$\big(|\mathcal{S}|, K, (\ell, c, h), \xi', r\big)$ ▷ larger $\xi' \Rightarrow$ lower variance

for $k = 1$ to K **do**

  $m_k \leftarrow L - |Y_k|$; $F_k \leftarrow$ SAMPLEWITHOUTREPLACEMENT$(\mathcal{U}, m_k)$

  $X_k \leftarrow Y_k \cup F_k$

end for

return $\{X_k\}_{k=1}^K$

C.2. Client Distributions under Partitioning Strategies

C.2.1. Preference Heterogeneity

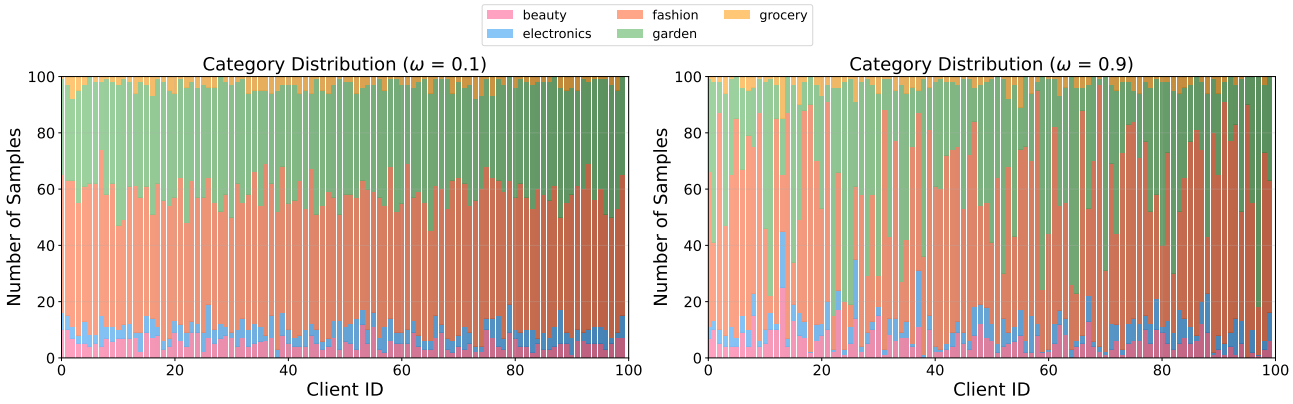


Figure 5: **Client Distribution under Preference Heterogeneity (WebShop,** $\omega = 0.1$ **vs.** $\omega = 0.9$**).**

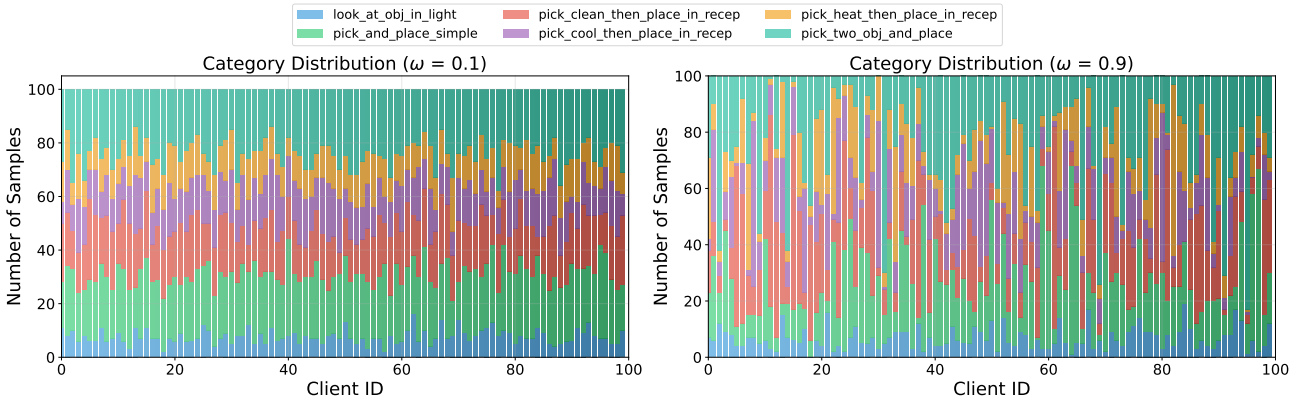


Figure 6: **Client Distribution under Preference Heterogeneity (ALFWorld,** $\omega = 0.1$ **vs.** $\omega = 0.9$**).**

C.2.2. COVERAGE HETEROGENEITY

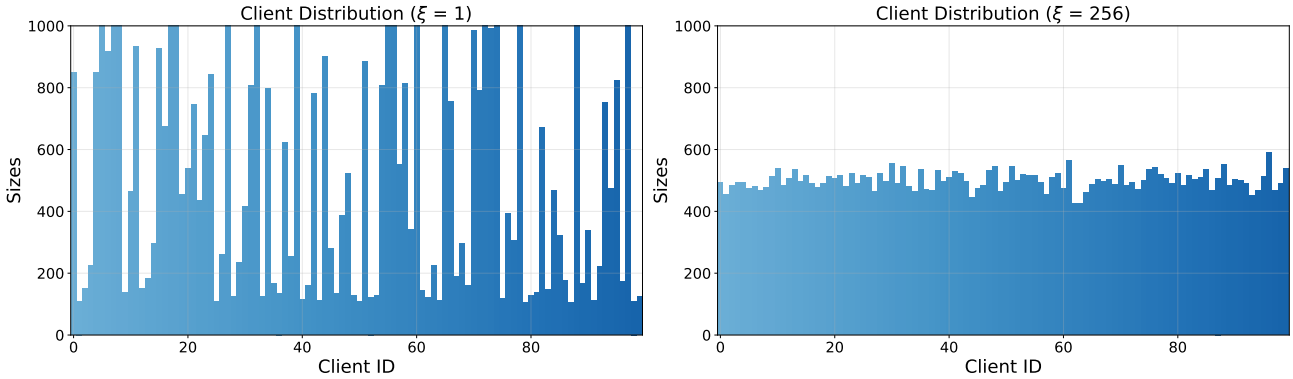


Figure 7: **Client Distribution under Coverage Heterogeneity (WebShop, $\xi = 1$ vs. $\xi = 256$).**

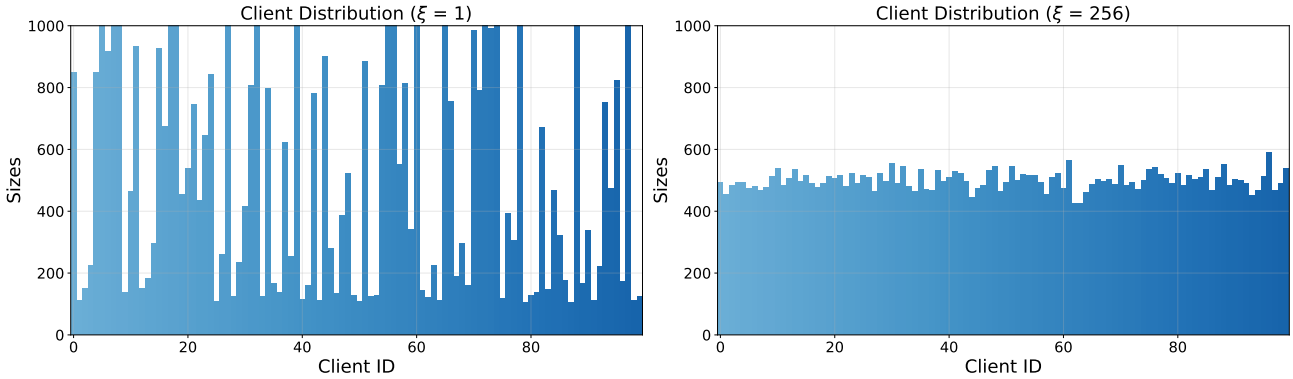


Figure 8: **Client Distribution under Coverage Heterogeneity (ALFWorld, $\xi = 1$ vs. $\xi = 256$).**

C.2.3. HARDNESS HETEROGENEITY

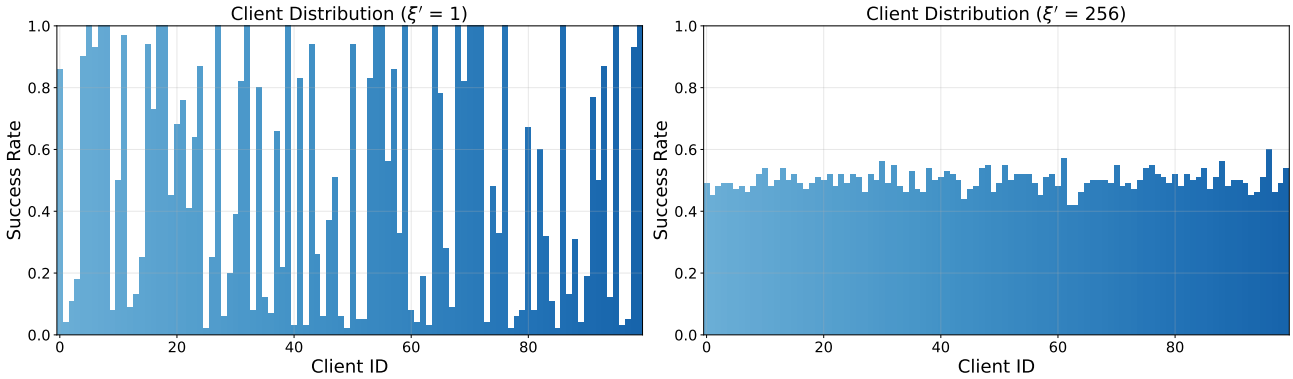


Figure 9: **Client Distribution under Hardness Heterogeneity (WebShop, $\xi' = 1$ vs. $\xi' = 256$).**

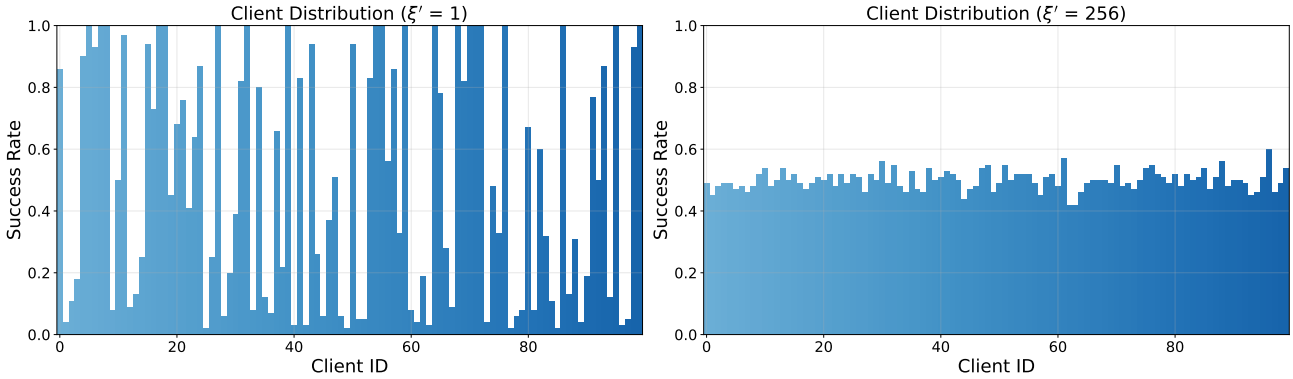


Figure 10: **Client Distribution under Hardness Heterogeneity (ALFWorld, $\xi' = 1$ vs. $\xi' = 256$).**