# MIND THE TOOL FAILURES: ACHIEVING SYNERGISTIC TOOL GAINS FOR MEDICAL AGENTS

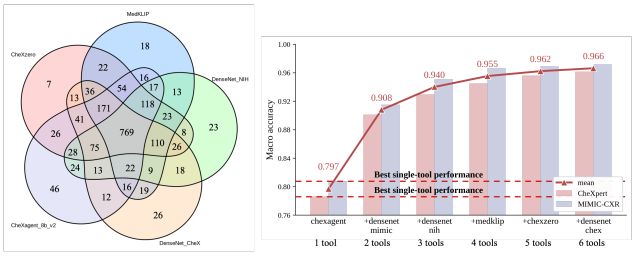
**Anonymous authors**
Paper under double-blind review

## ABSTRACT

For medical AI agents, disagreement among external tools can signal both a risk of error and the potential for better decisions: tools that fail on different instances may together outperform even the strongest individual tool. We define tool synergy as the ability to realize these complementary gains and characterize its potential through the Single-Oracle risk gap, which measures the performance gap between the best fixed single tool and an ideal instance-wise tool-use policy. Yet instance-level adaptation alone does not guarantee synergy, as standard outcome-based reinforcement learning and conventional aggregation methods do not explicitly exploit disagreement to learn when and how tools can complement one another. To address this challenge, we propose Collaborative Synergy Reinforcement Learning (CSRL), a GRPO-based framework that turns tool disagreement into an explicit learning signal for synergy. CSRL combines a Brier reward for probabilistic risk minimization, an override reward for disagreement-aware conflict resolution, and Entropy-Guided Sampling that prioritizes high-disagreement cases with stronger signals for learning complementary tool use. Across two tasks and seven medical benchmarks, CSRL consistently outperforms a broad range of baselines, with its advantage rising from 0.4% under tool consensus to 6.4% and 7.2% under moderate and high disagreement, respectively, highlighting its ability to turn tool disagreement into complementary gains.

## 1 INTRODUCTION

Recent advances in agentic systems have enabled language and multimodal models to select and use external tools for computation, retrieval, and specialized operations (Yao et al., 2022; Schick et al., 2023; Shen et al., 2023; Qian et al., 2025). However, task relevance does not guarantee instance-level reliability, as even appropriate tools can fail on individual cases. This issue is especially critical in medical agents, where tool failures may lead to unsafe downstream decisions (Habli et al., 2020; Fallahpour et al., 2025).




(a) Complementarity among imperfect medical tools (b) Marginal Oracle accuracy gains from adding tools on two datasets, CheXpert and MIMIC-CXR

Figure 1: (a) Analysis of tool complementarity, where each number indicates the number of instances correctly identified by the corresponding tools; (b) marginal Oracle gains on CheXpert and MIMIC-CXR.

Importantly, heterogeneous tool failures also imply potential complementarity. Different tools may succeed on different instances, allowing one tool to compensate for errors made by another. As shown in Figure 1(a), each tool correctly identifies instances missed by others, indicating that no single tool uniformly dominates across cases. To quantify this complementary potential, we further report the Oracle[1] accuracy on CheXpert (Irvin et al., 2019) and MIMIC-CXR (Johnson et al., 2019). As provided in Figure 1(b),

[1]An Oracle is an idealized evaluator that knows the correct answer and is used to estimate the best possible performance among the available candidates (Malmasi et al., 2015).



Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

the Oracle accuracy increases steadily as complementary tools are added and substantially exceeds the performance of the best individual tool, revealing considerable performance headroom from instance-level tool complementarity. In practice, this complementarity often appears as conflicting predictions among tools, making disagreement a useful signal for learning when and how their capabilities can be combined.

Yet instance-level adaptation alone does not guarantee tool synergy. Standard outcome-based reinforcement learning optimizes final-task correctness without explicitly learning how to resolve conflicting tool predictions or prioritizing cases with greater complementary potential, while conventional stacking and gating methods aggregate already-observed tool outputs rather than jointly learning which tools to invoke and how their conflicting evidence should influence the final decision. This motivates a central question: **how can an agent learn from tool disagreement to realize complementary gains beyond the strongest individual tool?**

To characterize the potential for realizing such tool synergy, we introduce the Single-Oracle risk gap, defined as the gap between the strongest fixed single-tool strategy and an ideal instance-wise tool-use policy. A positive gap quantifies the recoverable performance headroom created by complementary tool failures. Motivated by this risk gap, we propose **CSRL** (**C**ollaborative **S**ynergy **R**einforcement **L**earning), a GRPO-based framework for learning tool synergy from disagreement. Rather than treating disagreement merely as uncertainty to be resolved, CSRL uses it as a learning signal to guide both policy optimization and training-data allocation. Specifically, CSRL aligns learning at three complementary levels: a Brier reward encourages probabilistic risk minimization; an override reward provides decision-level supervision for resolving conflicting tool predictions by rewarding beneficial corrections of erroneous consensus while discouraging harmful overrides; and Entropy-Guided Sampling (EGS) increases exposure to high-disagreement cases, where differences across tools provide stronger signals for learning instance-specific synergy.

Through these mechanisms, CSRL learns to convert the complementary potential of the tool pool into realizable synergistic gains. Experiments across two tasks and seven medical benchmarks show that CSRL consistently outperforms individual tools, strong aggregation baselines, and Vanilla GRPO. Its advantage increases substantially with tool disagreement, from 0.4% under consensus to 6.4% and 7.2% under moderate and high disagreement, respectively.

Our contributions can be summarized as follows:

- We study tool synergy under heterogeneous failures from the perspective of tool disagreement, and introduce the Single-Oracle risk gap to characterize the potential for synergistic gains beyond the strongest individual tool.
- Guided by the Single-Oracle risk gap, we propose CSRL, a collaborative synergy reinforcement learning framework that learns to exploit tool disagreement through probabilistic risk minimization, disagreement-aware conflict resolution, and disagreement-focused sampling.
- Across two tasks and seven datasets, CSRL consistently outperforms individual tools, static and learned aggregation baselines, and both general and specialized multimodal large language models, with average gains of 7.5% in accuracy and 6.8% in F1.

## 2 METHOD

### 2.1 PRELIMINARY

**Problem Formulation.** Given a diagnostic sample $(x, q, y) \sim \mathcal{D}$, where $x \in \mathcal{X}$ is the input image, $q \in \mathcal{Q}$ is the target pathology query, and $y \in \mathcal{Y}$ is the ground-truth label, we consider a fixed pool of $K$ pretrained diagnostic tools $\mathcal{T} = \{T_1, \ldots, T_K\}$. For a given case, the policy may invoke a subset of tools from $\mathcal{T}$. For any invoked tool $T_k$, the final output of each tool can be represented as $p_k = T_k(x, q) \in \mathcal{P}$. Due to heterogeneity in model architectures and training data, these outputs may exhibit non-overlapping error patterns.

To formalize this complementarity, we first define the best single-tool risk $\mathcal{R}^{\star}_{\text{single}}$ as

$$\mathcal{R}^{\star}_{\text{single}} = \min_{k \in [K]} \mathbb{E}_{(x,q,y)\sim\mathcal{D}} \left[ \ell(p_k, y) \right], \tag{1}$$

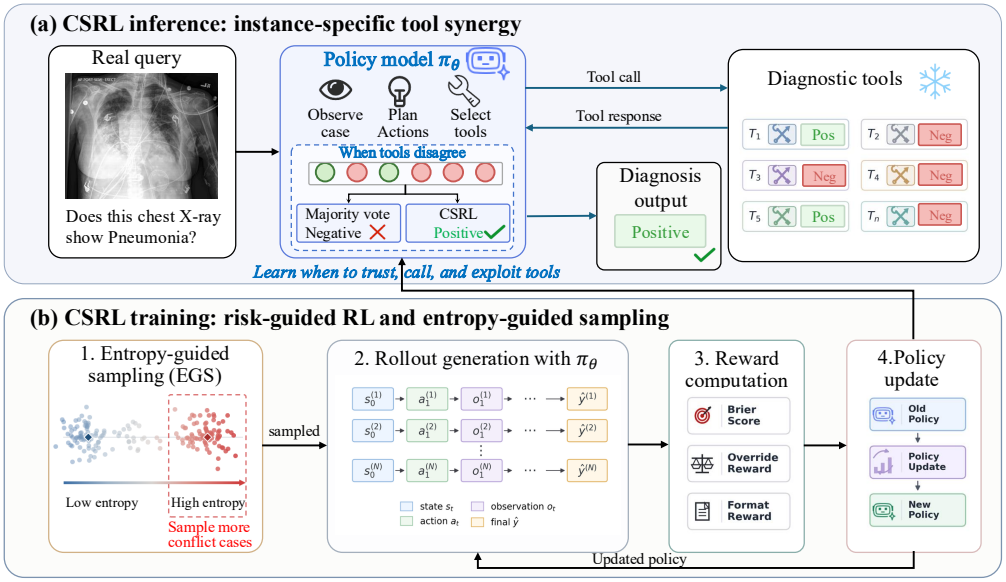



Figure 2: Overview of CSRL. (a) CSRL inference performs instance-specific agentic tool use. (b) CSRL training combines entropy-guided sampling with risk-guided reinforcement learning.

Where $\ell \in \mathcal{L} : \mathcal{P} \times \mathcal{Y} \to \mathbf{R}$ is a diagnostic loss, and $\mathcal{L}$ is the space of loss functions. As we discussed above, a single best tool is imperfect in real-world scenarios. Thus, we consider discovering case-level tool selection. Specifically, given $\Pi_\mathcal{T}$ be the space of tool integration policy $\pi : \mathcal{X} \times \mathcal{Q} \times \mathcal{P}^K \to \mathbf{R}$, the Oracle risk can be defined as

$$\mathcal{R}^\star_{\text{oracle}} = \inf_{\pi \in \Pi_\Theta} \mathbb{E}_{(x,q,y)\sim\mathcal{D}} \left[\ell(\pi(x, q, \mathbf{p}), y)\right]. \tag{2}$$

Since the single tool strategy $\pi_k(x, q, \mathbf{p}) = p_k$ is a special case in $\Pi_\mathcal{T}$, it is obvious that we can define the Single-Oracle risk gap as

$$\Delta_\mathcal{T} = \mathcal{R}^\star_{\text{single}} - \mathcal{R}^\star_{\text{oracle}} \geq 0. \tag{3}$$

A positive $\Delta_\mathcal{T}$ indicates that the strongest individual tool is not the performance limit of the tool pool. As shown in the Introduction, performance gains ($\Delta_\mathcal{T}$) are significantly large, which supports our assumption.

To optimize the model toward oracle-level performance, we aim to learn a parameterized tool-use policy $\pi_\theta \in \Pi_\mathcal{T}$. Given $\hat{p}_\theta = \pi_\theta(x, q, \mathbf{p})$, we aim to learn a policy satisfying

$$\mathcal{R}^\star_{\text{oracle}} \leq \mathcal{R}(\pi_\theta) = \mathbb{E}_{(x,q,y)\sim\mathcal{D}} \left[\ell(\hat{p}_\theta, y)\right] < \mathcal{R}^\star_{\text{single}}. \tag{4}$$

To this end, as provided in Figure 2, we propose CSRL, a Collaborative Synergy Reinforcement Learning framework built upon GRPO (Shao et al., 2024), which introduces two components that align the optimization procedure with the target in Eq. (4). First, we design trajectory-level rewards for GRPO that encourage valid tool use, minimize probabilistic diagnostic risk, and promote disagreement-aware synergy recovery, thereby directly pushing down $\mathcal{R}(\pi_\theta)$. Second, we use Entropy-Guided Sampling to increase the training mass of high-disagreement cases, where the gap between single-tool performance and case-specific integration is most recoverable.

**Tool-Calling Rollout.** Rollouts $\{\tau_i\}$ define the interaction protocol between the policy $\pi_\theta$ and the tool-augmented environment: the policy $\pi_\theta$ uses <tool_call>...</tool_call>to invoke diagnostic tools, receives tool feedback wrapped by <tool_response>...</tool_response>, and terminates the trajectory by producing a final prediction inside <answer>...</answer>. The detailed procedure is provided in Appendix G.

**GRPO Optimization.** We optimize $\pi_\theta$ with **Group Relative Policy Optimization** (GRPO) (Shao et al., 2024), using EGS-stratified mini-batches (Sec. 2.3) and the trajectory-level reward $R_{\text{overall}}$

from Sec. 2.2. For each prompt $(x, q)$, GRPO samples $G$ rollouts $\{\tau_i\}_{i=1}^{G}$ from $\pi_{\theta_{\text{old}}}$ and assigns each rollout a group-relative advantage

$$\hat{A}_i \;=\; \frac{R_{\text{overall}}(\tau_i) \;-\; \text{mean}\big(\{R_{\text{overall}}(\tau_j)\}_{j=1}^{G}\big)}{\text{std}\big(\{R_{\text{overall}}(\tau_j)\}_{j=1}^{G}\big)}, \tag{5}$$

which is shared across all policy-generated tokens in $\tau_i$. The GRPO objective combines a PPO-style clipped surrogate with a token-level KL penalty to a frozen reference policy $\pi_{\text{ref}}$:

$$\mathcal{J}_{\text{GRPO}}(\theta) \;=\; \mathbb{E}\left[\frac{1}{G}\sum_{i=1}^{G}\frac{1}{|\tau_i|}\sum_{t=1}^{|\tau_i|}\Big\{\min\big(\rho_{i,t}\hat{A}_i,\; \text{clip}(\rho_{i,t},\,1-\epsilon,\,1+\epsilon)\,\hat{A}_i\big) - \beta\,\mathbb{D}_{\text{KL}}\big[\pi_\theta\|\pi_{\text{ref}}\big]\Big\}\right], \tag{6}$$

with importance ratio

$$\rho_{i,t} = \frac{\pi_\theta(a_{i,t}\,|\,s_{i,t})}{\pi_{\theta_{\text{old}}}(a_{i,t}\,|\,s_{i,t})}. \tag{7}$$

### 2.2 Risk-guided Reward Design

To train the policy $\pi_\theta$ with GRPO, we assign each rollout a rule-based reward composed of three terms that capture format validity, probabilistic correctness, and case-specific majority override:

$$R_{\text{overall}} = R_{\text{brier}} + R_{\text{override}} + R_{\text{format}}. \tag{8}$$

To encourage a small $\mathcal{R}(\pi_\theta)$, we first introduce the Brier reward as follows:

$$R_{\text{brier}}(\hat{p}_\theta, y) = \begin{cases} 1-(\hat{p}_\theta - y)^2, & \text{if } \hat{p}_\theta \in [0,1] \text{ is parseable and } n_{\text{valid}} > 0, \\ 0, & \text{otherwise.} \end{cases} \tag{9}$$

Where $n_{\text{valid}}$ denotes the number of valid tool responses observed in the rollout. The Brier reward evaluates probabilistic correctness only when the rollout uses valid tool evidence. The following proposition characterizes the role of $R_{\text{brier}}$

**Proposition 2.1.** *Assume that the prediction risk is defined with the Brier loss, i.e.,*

$$\mathcal{R}(\pi_\theta) = \mathbb{E}_{(x,q,y)\sim\mathcal{D}}\left[(\hat{p}_\theta - y)^2\right].$$

*Then maximizing the Brier reward is equivalent to minimizing the prediction risk:*

$$\arg\max_\theta \mathbb{E}\left[R_{\text{brier}}(\hat{p}_\theta, y)\right] = \arg\min_\theta \mathcal{R}(\pi_\theta).$$

Proposition 2.1 illustrates that maximizing reward $R_{\text{brier}}$ is aligned with minimizing the risk $\mathcal{R}(\pi_\theta)$. $R_{\text{brier}}$ provides a proper probabilistic training signal for the final diagnostic score, but it is agnostic to the structure of tool disagreement. Such disagreement is central to the synergy gap, as it indicates that different tools succeed and fail on different cases, leaving room for a case-specific policy to recover information that no fixed tool can consistently capture.

To address the above issue, we introduce the override reward in the following.

$$R_{\text{override}}(\hat{p}_\theta, m, y) = \begin{cases} \alpha, & \text{if } \ell(\hat{p}_\theta, y) < s \text{ and } \ell(\hat{p}_m, y) \geq s, \\ -\alpha, & \text{if } \ell(\hat{p}_\theta, y) \geq s \text{ and } \ell(\hat{p}_m, y) < s, \\ 0, & \text{otherwise.} \end{cases} \tag{10}$$

Where $\hat{p}_m$ denotes the response of the majority vote among the valid invoked tools, $s$ is a threshold used to determine whether a prediction is consistent with the ground-truth label $y$ based on the loss $\ell$, e.g., $s = 0.5$ in classification tasks, and $\alpha$ is the reward value which is set as 0.1 in our experiments. The following proposition demonstrates how $R_{\text{override}}$ enlarges the risk gap.

**Proposition 2.2.** *Considering the risk $\mathcal{R}(\pi_\theta) = \Pr(\hat{p}_\theta \neq y)$ and the risk gap $\mathcal{G}(\pi_\theta) = \mathcal{R}^\star_{\text{single}} - \mathcal{R}(\pi_\theta)$, maximizing $R_{\text{override}}(\hat{p}_\theta, m, y)$ is aligned with maximizing $\mathcal{G}(\pi_\theta)$.*

Proposition 2.2 shows that the override reward is aligned with our objective of increasing $\mathcal{G}(\pi_\theta)$, thereby improving the policy beyond the best single-tool baseline. Proofs of Propositions 2.1 and 2.2 are provided in Appendix B.

In addition, to encourage the policy to follow the required structured output schema, we introduce a format reward as follows:

$$R_{\text{format}} = \begin{cases} \alpha, & \text{if all required fields appear in the correct order} \\ & \text{and the terminal answer is parseable into } [0,1], \\ 0, & \text{otherwise.} \end{cases} \tag{11}$$

In our experiments, we set $\alpha = 0.1$ in $R_{\text{format}}$. Although the reward design above is presented for classification, the underlying framework can be adapted to other tasks with task-specific reward functions.

### 2.3 Entropy-Guided Sampling

The risk gap $\Delta_{\mathcal{T}}$ is most visible in cases where the tool pool provides heterogeneous evidence. Under uniform sampling from the natural data distribution, these high-disagreement cases may be underrepresented, causing most GRPO groups to be dominated by consensus cases with weak signals for learning case-specific tool synergy. We therefore introduce Entropy-Guided Sampling (EGS), a static *per-batch* stratified sampler that reweights mini-batch construction toward high-entropy regions of the tool-evidence space. By increasing the frequency of disagreement cases during training, EGS provides denser policy-gradient signal for learning when and how to exploit complementary tools.

For each training case $(x, q, y)$, we record the number of correctly classifying tools:

$$n_c(x, q) = \sum_{k=1}^{K} \mathbb{I}\{\ell(\hat{p}_k, y) < s\}, \tag{12}$$

where $s$ is the threshold discussed in Eq. 10. We then measure the correctness-agreement entropy by

$$H(\bar{n}) = -\bar{n}\log\bar{n} - (1-\bar{n})\log(1-\bar{n}), \qquad \bar{n} = \frac{n_c}{K}. \tag{13}$$

This entropy quantifies case-level tool disagreement, reaching its maximum at $n_c = K/2$ and vanishing at $n_c \in \{0, K\}$. The useful learning signal for case-specific synergy is concentrated in the conflict region $1 \le n_c \le K-1$, where correct and incorrect tools coexist. In contrast, consensus cases provide limited synergy signal: $n_c = 0$ suggests either label noise or cases outside the capability of the tool pool, whereas $n_c = K$ indicates that all tools are already correct and case-specific aggregation cannot further improve the prediction.

We therefore stratify the training set according to the entropy-based disagreement level. By the symmetry of the binary entropy function in Eq. (13), $H(n_1/K) = H(n_2/K)$ when $n_1/K = 1 - n_2/K$. Consequently, the $K+1$ possible correct-tool counts can be grouped into $K'$ subsets according to their entropy values. For example, when $K = 6$, the seven possible counts from 0 to 6 form $K' = 4$ subsets, with entropy levels $H_{S_1} = H(0/6)$, $H_{S_2} = H(1/6)$, $H_{S_3} = H(2/6)$, and $H_{S_4} = H(3/6)$.

We assign a small nonzero sampling probability $\epsilon$ to the zero-entropy consensus subset $S_1$, while allocating the remaining probability mass among the non-consensus subsets in proportion to their entropy values:

$$\rho_1 = \epsilon, \qquad \rho_j = (1-\epsilon)\frac{H_{S_j}}{\sum_{\ell=2}^{K'} H_{S_\ell}}, \quad j = 2, \ldots, K'. \tag{14}$$

Here, $H_{S_j} = H((j-1)/K)$. In our implementation, we set

$$\epsilon = \frac{2H(1/K)}{K}. \tag{15}$$

This choice guarantees $\rho_1 < \rho_2$, such that the consensus subset is sampled less frequently than the lowest-entropy non-consensus subset while retaining a strictly positive sampling probability. Thus, higher-entropy subsets are sampled more frequently, while the consensus subset retains a nonzero probability to preserve calibration on easy and agreement cases. The detailed derivation is provided in Appendix B.2.

# 3 EXPERIMENTS

## 3.1 EXPERIMENTAL SETUP

**Datasets.** We conduct experiments on two tasks and seven datasets on the chest X-ray (CXR) modality. For *binary pathology classification* (Task 1), CSRL is trained on the official training splits of CheXpert (Irvin et al., 2019) and MIMIC-CXR (Johnson et al., 2019), and evaluated on six test benchmarks: their held-out test splits, three stratified $1{,}000$-image out-of-domain samples from NIH ChestX-ray14 (Wang et al., 2017), RSNA Pneumonia Detection (Shih et al., 2019), and VinDr-CXR (Nguyen et al., 2022), plus NIH-Google (Majkowska et al., 2020) ($1{,}962$ radiologist-adjudicated images) as a clean-label check against NIH NLP label noise. For *medical VQA generalization* (Task 2), the policy is trained on a stratified $5{,}000$-sample subset of the *choose_train* split of MIMIC-Ext-MIMIC-CXR-VQA (Bae et al., 2023; 2024) under the M4CXR non-empty-GT setting, and evaluated on the gold $1{,}523$-question test split and its $933$-question non-empty-GT subset (Park et al., 2025).

**Policy Model and Tool Pools.** We use Qwen2.5-VL-7B-Instruct (Bai et al., 2025b) as the policy model for reinforcement learning. We assemble two task-specific tool pools chosen for diversity in training data and architecture. For Task 1 (*binary pathology classification*), the tool pool includes six specialist tools, namely CheXagent (Chen et al., 2024d), CheXzero (Tiu et al., 2022), MedKLIP (Wu et al., 2023a), and three DenseNet-121 backbones from torchxrayvision (Cohen et al., 2022). For Task 2 (*VQA generalization*), the tool pool includes three medical vision-language models, namely CheXagent-8B (Chen et al., 2024d), RadVLM-7B (Deperrois et al., 2025), and MedGemma-4B-IT (Sellergren et al., 2025).

**Implementation Details.** We implement CSRL with the Qwen2.5-VL-7B-Instruct (Bai et al., 2025b) backbone. The policy is optimized using the Group Relative Policy Optimization (GRPO) algorithm (Shao et al., 2024; Guo et al., 2025) for $\sim 75$ iterations on 8 NVIDIA H200 GPUs, built on VERL (Sheng et al., 2025) with FSDP training and asynchronous vLLM (Kwon et al., 2023) rollouts. During training, we use a batch size of $256$ prompts; for each prompt, the agent samples $16$ trajectories with up to $4$ reasoning turns and a maximum of $6$ parallel tool calls per turn. Key hyperparameters are AdamW with learning rate $1\times10^{-6}$, PPO clip ratio $0.2$, low-variance KL regularization with coefficient $10^{-3}$ against a frozen reference policy, and an entropy bonus of $10^{-3}$. Additional training details are provided in Appendix H.

**Baselines and Comparisons.** We compare CSRL with a broad set of baselines spanning four categories: open-source general vision-language models (VLMs), open-source medical MLLMs, closed-source general MLLMs, and specialist CXR diagnostic tools. To assess whether the gains arise from case-specific tool synergy rather than simple tool combination, we further include both static and learned ensemble/aggregation baselines. Static baselines include logistic regression, majority voting, and best-tool-per-pathology selection, while learned baselines include Frozen and SFT MLLM Aggregators, stacking methods (Wolpert, 1992) based on MLP, Random Forest (Breiman, 2001), XGBoost (Chen & Guestrin, 2016), and LightGBM (Ke et al., 2017), and Image-Aware MLP Stacking.

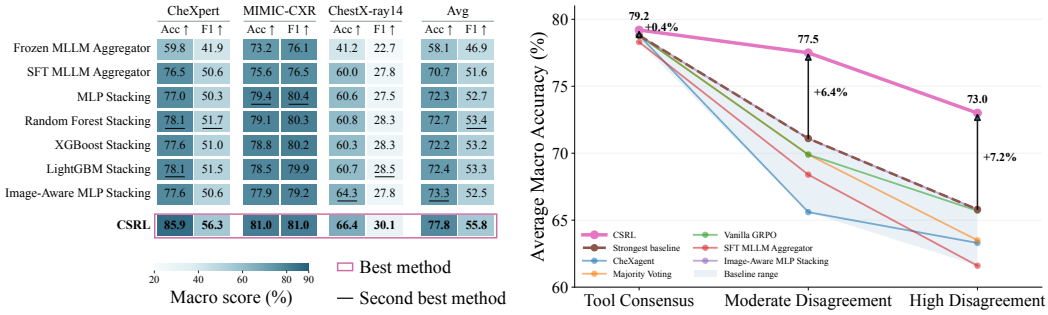
## 3.2 EXPERIMENTAL ANALYSIS

To systematically evaluate the effectiveness of our proposed approach, we organize the experiment as follows.

**Learning tool synergy yields robust gains across in-domain and out-of-domain datasets.** Our proposed CSRL achieves the best overall performance among all compared methods (see Table 1), obtaining an average Acc of 82.2% and an average F1 of 57.8% across the six CXR benchmarks, outperforming the strongest prior method by 3.3% in accuracy and 4.8% in F1 score. Compared with the strongest individual tool, CheXzero, CSRL improves the average performance by **7.5%** in Acc and **6.8%** in F1, showing that explicitly learning tool synergy is more effective than relying on any single tool. This advantage is also consistent across datasets: CSRL almost achieves the best or the second-best F1 score and Acc on all six benchmarks. Notably, CSRL performs strongly on both in-domain and out-of-domain test sets, including ChestX-ray14, VinDr-CXR, NIH-Google, and RSNA, suggesting that learned tool synergy helps the model better exploit complementary diagnostic capabilities under distribution shifts. In contrast, individual models and both static and learned tool-combination baselines achieve lower overall performance. These results demonstrate that CSRL

Table 1: Performance comparison for binary disease diagnosis across six chest X-ray benchmarks. We report macro accuracy (Acc) and macro F1. Out-of-domain (OOD) test sets are marked with ◇. Best results are highlighted in **bold**, and second-best results are underlined.

| Methods | CheXpert | | MIMIC-CXR | | ChestX-ray14◇ | | VinDr-CXR◇ | | NIH-Google◇ | | RSNA◇ | | Average | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Acc↑ | F1↑ | Acc↑ | F1↑ | Acc↑ | F1↑ | Acc↑ | F1↑ | Acc↑ | F1↑ | Acc↑ | F1↑ | Acc↑ | F1↑ |
| **Open-source General VLMs** | | | | | | | | | | | | | | |
| Qwen2.5-VL-7B-Instruct (Bai et al., 2025b) | 73.1 | 18.1 | 53.9 | 37.1 | 72.2 | 15.5 | 81.7 | 12.1 | 79.3 | 37.8 | 77.1 | 33.6 | 72.9 | 25.7 |
| InternVL3.5-38B-Instruct (Wang et al., 2025) | 74.4 | 45.1 | 79.0 | 71.0 | 58.5 | 22.8 | 74.6 | 34.0 | 87.9 | 47.8 | **79.0** | 62.8 | 75.6 | 47.2 |
| Qwen2.5-VL-72B-Instruct (Bai et al., 2025b) | 38.2 | 30.9 | 71.3 | 70.3 | 37.5 | 17.4 | 43.5 | 10.0 | 78.5 | 42.9 | 42.5 | 42.9 | 51.9 | 35.7 |
| **Open-source Medical MLLMs** | | | | | | | | | | | | | | |
| HuatuoGPT-Vision-7B (Chen et al., 2024b) | 69.9 | 39.7 | 69.8 | 66.7 | 51.2 | 21.2 | 89.4 | 28.2 | 84.7 | 53.5 | 72.6 | 58.0 | 72.9 | 44.6 |
| Lingshu-7B (Xu et al., 2025) | 53.4 | 38.6 | 71.1 | 75.1 | 53.2 | 25.9 | 77.0 | 24.8 | 72.3 | 59.7 | 70.6 | 57.5 | 66.3 | 46.9 |
| MedGemma-27B-IT (Sellergren et al., 2025) | 72.2 | 45.4 | 72.1 | 73.3 | 62.0 | 28.0 | 90.2 | 40.3 | 77.7 | 61.5 | 74.2 | 58.5 | 74.7 | 51.2 |
| Hulu-Med-Flash-Preview-27B (Jiang et al., 2025) | 72.8 | 25.2 | 76.7 | 76.7 | **72.7** | 27.7 | 86.8 | 41.1 | 85.4 | 66.6 | **79.0** | 29.1 | 78.9 | 44.4 |
| **Closed-source General MLLMs** | | | | | | | | | | | | | | |
| Claude-Sonnet-4.6 | 47.3 | 33.7 | 70.9 | 70.8 | 33.0 | 19.2 | 54.3 | 12.9 | 75.4 | 52.3 | 46.5 | 43.3 | 54.6 | 38.7 |
| GPT-4o | 58.2 | 35.5 | 69.9 | 67.2 | 49.9 | 20.4 | 85.1 | 20.1 | 84.5 | 50.9 | 68.7 | 55.6 | 69.4 | 41.6 |
| Gemini-3.1-Pro-Preview | 68.5 | 43.6 | 76.9 | 76.0 | 53.5 | 25.4 | 93.0 | 45.5 | 84.9 | 67.0 | 73.1 | 60.2 | 75.0 | 53.0 |
| **Specialist CXR Tools** | | | | | | | | | | | | | | |
| DenseNet-CheX (Cohen et al., 2022) | 68.2 | 43.3 | 70.4 | 71.1 | 51.7 | 22.5 | 91.6 | 27.1 | 70.2 | 52.2 | 68.6 | 50.5 | 70.1 | 44.5 |
| DenseNet-MIMIC (Cohen et al., 2022) | 65.7 | 41.3 | 70.1 | 68.6 | 55.9 | 20.8 | 78.7 | 17.9 | 77.4 | 49.1 | 70.3 | 46.1 | 69.7 | 40.6 |
| DenseNet-NIH (Cohen et al., 2022) | 60.7 | 37.9 | 66.5 | 68.8 | 44.0 | 21.8 | 73.2 | 15.7 | 61.1 | 50.0 | 68.9 | 49.8 | 62.4 | 40.7 |
| CheXzero (Tiu et al., 2022) | 76.3 | 48.9 | 74.6 | 75.6 | 60.3 | 27.4 | 90.4 | 36.2 | 78.1 | 61.4 | 68.3 | 56.3 | 74.7 | 51.0 |
| MedKLIP (Wu et al., 2023a) | 71.3 | 46.4 | 73.7 | 76.2 | 58.0 | 26.1 | 89.7 | 38.3 | 67.7 | 56.5 | 71.4 | 58.9 | 72.0 | 50.4 |
| CheXagent-8B (Chen et al., 2024d) | 78.6 | 49.8 | 80.8 | 80.2 | 50.7 | 26.3 | 50.2 | 22.5 | 74.6 | 61.4 | 49.2 | 45.1 | 64.0 | 47.6 |
| **Tool Combination Baselines** | | | | | | | | | | | | | | |
| Logistic Regression | 78.4 | 51.0 | 79.1 | 80.5 | 58.6 | 25.6 | 82.1 | 36.9 | 85.3 | 64.9 | 45.9 | 44.6 | 71.6 | 50.6 |
| Majority Voting (tie → neg) | 76.6 | 49.0 | 77.4 | 76.2 | 60.8 | 27.0 | 93.4 | 40.9 | 83.0 | 63.4 | 74.5 | 60.2 | 77.6 | 52.8 |
| Majority Voting (tie → pos) | 69.6 | 46.7 | 77.9 | 77.9 | 49.4 | 24.9 | 87.5 | 33.3 | 73.1 | 58.6 | 68.2 | 56.9 | 70.9 | 49.7 |
| Best Tool per Pathology | 80.1 | 51.6 | 80.4 | **81.2** | 61.7 | 27.8 | 82.2 | 37.4 | 86.6 | 67.4 | 49.2 | 45.1 | 73.4 | 51.8 |
| **Ours (CSRL)** | **85.9** | **56.3** | **81.0** | 81.0 | 66.4 | **30.1** | **93.5** | **48.2** | **88.9** | **68.2** | 77.2 | **63.0** | **82.2** | **57.8** |




(a) Ensemble/aggregation comparison.

(b) Tool disagreement analysis.

Figure 3: Comparison with ensemble/aggregation baselines and analysis under different levels of tool disagreement. (a) Performance on CheXpert, MIMIC-CXR, and ChestX-ray14. (b) Conflict-specific performance under increasing tool disagreement.

can learn when and how different tools complement each other, leading to more robust and accurate binary disease diagnosis than the best individual tool.

**CSRL outperforms ensemble/aggregation baselines.** Figure 3a compares CSRL with a range of learned ensemble and aggregation methods on CheXpert, MIMIC-CXR, and ChestX-ray14. CSRL achieves the strongest average performance, reaching 77.8% Macro Accuracy and 55.8% Macro F1, compared with 73.3% Accuracy and 53.4% F1 for the strongest ensemble/aggregation baselines. The gains are also consistent across the three datasets, showing that simply learning to aggregate tool outputs is insufficient to recover the complementary information captured by CSRL.

Beyond binary disease diagnosis, Table 2 further evaluates the generalization ability of CSRL on the VQA task. CSRL consistently achieves the best performance across all metrics. Compared with the strongest baseline, MedGemma-4B-IT, CSRL improves the average score from 65.7% to 75.3%, corresponding to a gain of 9.6%, indicating that learning tool synergy not only improves classification

performance but also enhances the model's ability to recover more complete and accurate answers in a different task setting.

**CSRL gains increase with tool disagreement.** We retrospectively group test instances by $n_c$, the number of correct tools among the six available tools, into Tool Consensus ($n_c \in \{0, 6\}$), Moderate Disagreement ($n_c \in \{1, 5\}$), and High Disagreement ($n_c \in \{2, 3, 4\}$). The two disagreement groups account for 67.2% of the test instances. Figure 3b reports the average Macro Accuracy across CheXpert, MIMIC-CXR, and ChestX-ray14. CSRL achieves the highest Macro Accuracy in all three groups, and its advantage over the strongest baseline increases from 0.4% under Tool Consensus to 6.4% and 7.2% under Moderate and High Disagreement, respectively. This widening gap shows that CSRL provides larger gains when tool evidence becomes more conflicting. Additional recovery and override analyses are provided in Appendix F. We further examine whether these improvements are concentrated in particular patient subgroups through a retrospective demographic analysis in Appendix E.

Table 2: Generalization performance on the VQA task. We report F1, exact match (EM), precision (P), recall (R), and their average score. Best results are highlighted in **bold**.

| Model | F1↑ | EM↑ | P↑ | R↑ | Avg.↑ |
|---|---|---|---|---|---|
| Qwen2.5-VL-7B-Instruct | 41.7 | 26.8 | 43.1 | 43.9 | 38.9 |
| MedGemma-4B-IT | 69.1 | 50.9 | 71.2 | 71.5 | 65.7 |
| RadVLM-7B | 65.3 | 52.8 | 69.8 | 63.8 | 62.9 |
| CheXagent-8B | 68.2 | 54.5 | 75.0 | 64.7 | 65.6 |
| **CSRL (ours)** | **78.5** | **61.1** | **81.6** | **79.8** | **75.3** |

### 3.3 Ablation Study

**Distinct roles of reward design and entropy-guided sampling.** Table 3 further separates the effects of task reward, override reward, and sampling strategy. EGS consistently improves average Macro Accuracy regardless of the reward configuration, increasing performance from 72.5% to 74.5% with the ordinary accuracy reward, from 74.7% to 76.8% with the Brier reward, and from 75.6% to 77.8% when the override reward is also used. The override reward provides an additional improvement under both sampling strategies, increasing average Macro Accuracy from 74.7% to 75.6% under uniform sampling and from 76.8% to 77.8% with EGS. The corresponding F1 improvements are also larger when EGS and the override reward are combined, with the full CSRL achieving the best overall performance of 77.8% Acc and 55.8% F1. These results show that EGS controls which training instances are emphasized, whereas the override reward shapes how conflicting tool evidence is resolved. Their complementary effects lead to the strongest overall performance.

Table 3: Controlled ablation of reward design and entropy-guided sampling. Results are averaged across CheXpert, MIMIC-CXR, and ChestX-ray14.

| Method | Task Reward | Override | Sampling | Avg. Acc | Avg. F1 |
|---|---|---|---|---|---|
| Vanilla GRPO | Accuracy | × | Uniform | 72.5 | 52.7 |
| Vanilla GRPO + EGS | Accuracy | × | EGS | 74.5 | 52.7 |
| CSRL w/o Override Reward & EGS | Brier | × | Uniform | 74.7 | 50.8 |
| CSRL w/o Override Reward | Brier | × | EGS | 76.8 | 53.9 |
| CSRL w/o EGS | Brier | ✓ | Uniform | 75.6 | 51.3 |
| **CSRL** | **Brier** | ✓ | **EGS** | **77.8** | **55.8** |

**Scalability across training data, policy models, and tool budgets.** Figure 4 summarizes how CSRL scales with the number of available tools, policy model capacity, and training data, with detailed per-dataset results provided in Appendix C. Increasing the number of tools generally improves performance, but the marginal gain diminishes as the tool set becomes larger. As shown in the tool scalability ablation (Figure 4d), expanding the tool set from T1 to T3 brings a substantial improvement, with average macro Acc increasing from 70.3% to 77.6% and average macro F1 increasing from 52.4% to 54.3%. However, further increasing the tool set from T3 to T6 yields only limited additional gains, with Acc fluctuating around 77%–78% and F1 improving more moderately to 55.8%. This suggests that a small number of complementary tools can already capture most of the useful inter-tool synergy, while adding more tools may introduce redundancy and higher inference cost.

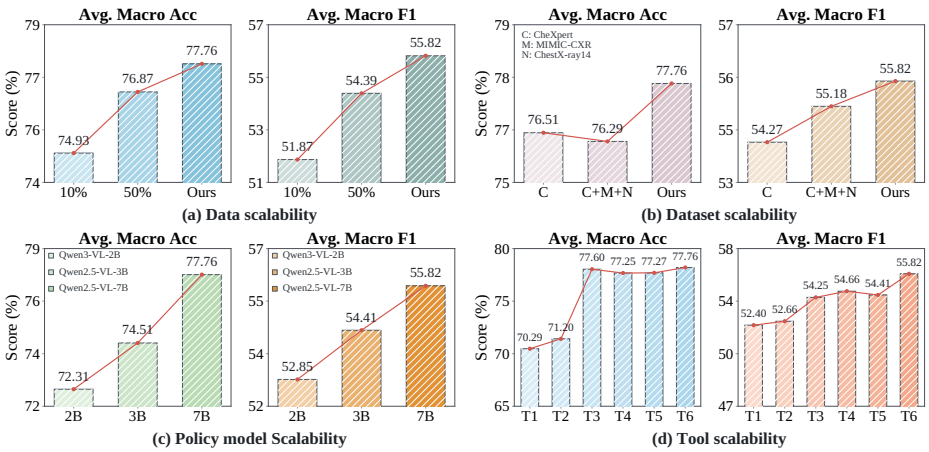



Figure 4: Scalability studies of CSRL. (a) Scalability across training data ratios; (b) Scalability across different dataset compositions; (c) Scalability across various policy models. (d) Scalability across tool-set sizes. T1-T6 denote tool sets with 1 to 6 tools, respectively.

The results also show that performance is affected not only by the number of tools, but also by the capacity of the policy model (Figure 4c) and the scale and composition of the training data (Figures 4a and 4b). Larger policy models consistently improve performance, from 72.3% Acc and 52.9% F1 with the 2B model to 77.8% Acc and 55.8% F1 with the 7B model, indicating that stronger policy models are better at learning when different tools should be trusted or combined. Similarly, increasing the training data scale improves both Acc and F1, while the dataset-composition results show that simply adding more training data does not necessarily improve generalization. Overall, these results indicate that the best performance-cost balance is achieved by selecting a compact but complementary tool set and pairing it with a sufficiently capable policy model, rather than naively increasing the number of tools.

## 4 Related Work

Tool-augmented agents extend language and multimodal models with external tools, APIs, and specialized models, with prior work focusing on tool selection, planning, and reinforcement-learning-based tool use (Yao et al., 2022; Schick et al., 2023; Shen et al., 2023; Qian et al., 2025). In medical AI, specialized diagnostic models and medical vision-language models provide useful external signals for radiology interpretation and multimodal reasoning (Fallahpour et al., 2025; Tiu et al., 2022; Wu et al., 2023a; Chen et al., 2024c). Our work instead focuses on settings where multiple task-relevant tools exhibit heterogeneous, instance-dependent failures. Rather than relying on a single tool or conventional aggregation, CSRL explicitly learns from tool disagreement to exploit case-level complementarity and achieve synergistic gains beyond the strongest individual tool. A more comprehensive discussion is provided in Appendix A.

## 5 Conclusion

Motivated by the challenge of turning the complementary potential of imperfect tools into realizable gains, we presented CSRL (Collaborative Synergy Reinforcement Learning), a GRPO-based framework that learns tool synergy by exploiting disagreement among tools. Guided by the Single-Oracle risk gap, CSRL aligns probabilistic risk minimization, disagreement-aware conflict resolution, and disagreement-focused sampling to achieve performance beyond the strongest individual tool. Across chest X-ray diagnosis and medical VQA, CSRL consistently outperforms individual tools, static and learned aggregation baselines, and general or medical multimodal models, with larger gains as tool disagreement increases. These results provide evidence that explicitly learning from tool disagreement offers an effective way to convert complementary tool capabilities into performance gains beyond standalone tools.

### AI USE STATEMENT

We used ChatGPT and Codex (OpenAI) to polish the language of the manuscript, recommend and summarize relevant literature, compare related methods, assist with code implementation, and support the analysis and interpretation of experimental results. The authors reviewed all AI-assisted text revisions and code, verified the original sources and bibliographic details of AI-recommended references included in the paper, and checked that these references support the associated claims. AI-assisted analyses were verified against the actual experimental outputs. The authors take full responsibility for the final content of this work, including its code, technical claims, experimental results, and scientific conclusions.

### ETHICS STATEMENT

This study uses existing chest X-ray and medical visual question answering datasets and does not involve the collection of additional patient data or prospective clinical intervention. The goal of this work is to improve the reliability of medical tool-use agents when individual diagnostic tools produce incorrect or conflicting predictions. However, the proposed system may inherit biases, annotation errors, and dataset-specific limitations from the underlying datasets and pretrained tools, and its performance may vary across patient populations, institutions, and clinical settings. Errors or overreliance on automated outputs could lead to inappropriate clinical decisions. Therefore, the reported benchmark results should not be interpreted as evidence of clinical safety or readiness for deployment. Any real-world clinical use would require further validation, careful monitoring, and oversight by qualified healthcare professionals.

## REPRODUCIBILITY STATEMENT

Section 2 describes the CSRL framework, including the optimization objective, reward design, and entropy-guided sampling strategy. Section 3.1 specifies the datasets, training and evaluation splits, policy model, tool pools, and implementation settings. Appendix B presents the derivations of the theoretical propositions, while Appendix G details the tool-calling rollout procedure. Appendix H provides the software versions and system prompt used for chest X-ray classification, and Table 10 reports the training hyperparameters and hardware configuration.

## REFERENCES

Seongsu Bae, Daeun Kyung, Jaehee Ryu, Eunbyeol Cho, Gyubok Lee, Sunjun Kweon, Jungwoo Oh, Lei Ji, Eric Chang, Tackeun Kim, et al. Ehrxqa: A multi-modal question answering dataset for electronic health records with chest x-ray images. *Advances in Neural Information Processing Systems*, 36:3867–3880, 2023.

Seongsu Bae, Daeun Kyung, Jaehee Ryu, Eunbyeol Cho, Gyubok Lee, Sunjun Kweon, Jungwoo Oh, Lei JI, Eric Chang, Tackeun Kim, et al. Mimic-ext-mimic-cxr-vqa: A complex diverse and large-scale visual question answering dataset for chest x-ray images. *PhysioNet*, 2024.

Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, Wei Ding, Chang Gao, Chunjiang Ge, et al. Qwen3-vl technical report. *arXiv preprint arXiv:2511.21631*, 2025a.

Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, Humen Zhong, Yuanzhi Zhu, Mingkun Yang, Zhaohai Li, Jianqiang Wan, Pengfei Wang, Wei Ding, Zheren Fu, Yiheng Xu, Jiabo Ye, Xi Zhang, Tianbao Xie, Zesen Cheng, Hang Zhang, Zhibo Yang, Haiyang Xu, and Junyang Lin. Qwen2.5-vl technical report. *ArXiv*, abs/2502.13923, 2025b. URL https://api.semanticscholar.org/CorpusID:276449796.

Leo Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.

Junying Chen, Chi Gui, Ruyi Ouyang, Anningzhe Gao, Shunian Chen, Guiming Hardy Chen, Xidong Wang, Zhenyang Cai, Ke Ji, Xiang Wan, et al. Towards injecting medical visual knowledge into

multimodal llms at scale. In *Proceedings of the 2024 conference on empirical methods in natural language processing*, pp. 7346–7370, 2024a.

Junying Chen, Ruyi Ouyang, Anningzhe Gao, Shunian Chen, Guiming Hardy Chen, Xidong Wang, Ruifei Zhang, Zhenyang Cai, Ke Ji, Guangjun Yu, Xiang Wan, and Benyou Wang. Huatuogpt-vision, towards injecting medical visual knowledge into multimodal llms at scale, 2024b. URL https://arxiv.org/abs/2406.19280.

Tianqi Chen and Carlos Guestrin. Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, pp. 785–794, 2016.

Zhihong Chen, Maya Varma, Jean-Benoit Delbrouck, Magdalini Paschali, Louis Blankemeier, Dave Van Veen, Jeya Maria Jose Valanarasu, Alaa Youssef, Joseph Paul Cohen, Eduardo Pontes Reis, et al. Chexagent: Towards a foundation model for chest x-ray interpretation. In *AAAI 2024 Spring Symposium on Clinical Foundation Models*, 2024c.

Zhihong Chen, Maya Varma, Justin Xu, Magdalini Paschali, Dave Van Veen, Andrew Johnston, Alaa Youssef, Louis Blankemeier, Christian Bluethgen, Stephan Altmayer, et al. A vision-language foundation model to enhance efficiency of chest x-ray interpretation. *arXiv preprint arXiv:2401.12208*, 2024d.

Joseph Paul Cohen, Joseph D Viviano, Paul Bertin, Paul Morrison, Parsa Torabian, Matteo Guarrera, Matthew P Lungren, Akshay Chaudhari, Rupert Brooks, Mohammad Hashir, et al. Torchxrayvision: A library of chest x-ray datasets and models. In *International Conference on Medical Imaging with Deep Learning*, pp. 231–249. PMLR, 2022.

Nicolas Deperrois, Hidetoshi Matsuo, Samuel Ruipérez-Campillo, Moritz Vandenhirtz, Sonia Laguna, Alain Ryser, Koji Fujimoto, Mizuho Nishio, Thomas M Sutter, Julia E Vogt, et al. Radvlm: a multitask conversational vision-language model for radiology. *arXiv preprint arXiv:2502.03333*, 2025.

Adibvafa Fallahpour, Jun Ma, Alif Munim, Hongwei Lyu, and Bo Wang. Medrax: Medical reasoning agent for chest x-ray. *arXiv preprint arXiv:2502.02673*, 2025.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, et al. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025.

Ibrahim Habli, Tom Lawton, and Zoe Porter. Artificial intelligence in health care: accountability and safety. *Bulletin of the World Health Organization*, 98(4):251, 2020.

Jeremy Irvin, Pranav Rajpurkar, Michael Ko, Yifan Yu, Silviana Ciurea-Ilcus, Chris Chute, Henrik Marklund, Behzad Haghgoo, Robyn Ball, Katie Shpanskaya, et al. Chexpert: A large chest radiograph dataset with uncertainty labels and expert comparison. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pp. 590–597, 2019.

Songtao Jiang, Yuan Wang, Sibo Song, Tianxiang Hu, Chenyi Zhou, Bin Pu, Yan Zhang, Zhibo Yang, Yang Feng, Joey Tianyi Zhou, et al. Hulu-med: A transparent generalist model towards holistic medical vision-language understanding. *arXiv preprint arXiv:2510.08668*, 2025.

Bowen Jin, Hansi Zeng, Zhenrui Yue, Jinsung Yoon, Sercan Arik, Dong Wang, Hamed Zamani, and Jiawei Han. Search-r1: Training llms to reason and leverage search engines with reinforcement learning. *arXiv preprint arXiv:2503.09516*, 2025.

Alistair EW Johnson, Tom J Pollard, Seth J Berkowitz, Nathaniel R Greenbaum, Matthew P Lungren, Chih-ying Deng, Roger G Mark, and Steven Horng. Mimic-cxr, a de-identified publicly available database of chest radiographs with free-text reports. *Scientific data*, 6(1):317, 2019.

Guolin Ke, Qi Meng, Thomas Finley, Taifeng Wang, Wei Chen, Weidong Ma, Qiwei Ye, and Tie-Yan Liu. Lightgbm: A highly efficient gradient boosting decision tree. *Advances in neural information processing systems*, 30, 2017.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph Gonzalez, Hao Zhang, and Ion Stoica. Efficient memory management for large language model serving with pagedattention. In *Proceedings of the 29th symposium on operating systems principles*, pp. 611–626, 2023.

Minghao Li, Yingxiu Zhao, Bowen Yu, Feifan Song, Hangyu Li, Haiyang Yu, Zhoujun Li, Fei Huang, and Yongbin Li. Api-bank: A comprehensive benchmark for tool-augmented llms. In *Proceedings of the 2023 conference on empirical methods in natural language processing*, pp. 3102–3116, 2023.

Anna Majkowska, Sid Mittal, David F Steiner, Joshua J Reicher, Scott Mayer McKinney, Gavin E Duggan, Krish Eswaran, Po-Hsuan Cameron Chen, Yun Liu, Sreenivasa Raju Kalidindi, et al. Chest radiograph interpretation with deep learning models: assessment with radiologist-adjudicated reference standards and population-adjusted evaluation. *Radiology*, 294(2):421–431, 2020.

Shervin Malmasi, Joel Tetreault, and Mark Dras. Oracle and human baselines for native language identification. In *Proceedings of the tenth workshop on innovative use of NLP for building educational applications*, pp. 172–178, 2015.

Ha Q Nguyen, Khanh Lam, Linh T Le, Hieu H Pham, Dat Q Tran, Dung B Nguyen, Dung D Le, Chi M Pham, Hang TT Tong, Diep H Dinh, et al. Vindr-cxr: An open dataset of chest x-rays with radiologist's annotations. *Scientific Data*, 9(1):429, 2022.

Bhargavi Paranjape, Scott Lundberg, Sameer Singh, Hannaneh Hajishirzi, Luke Zettlemoyer, and Marco Tulio Ribeiro. Art: Automatic multi-step reasoning and tool-use for large language models. *arXiv preprint arXiv:2303.09014*, 2023.

Jonggwon Park, Soobum Kim, Byungmu Yoon, Jihun Hyun, and Kyoyun Choi. M4cxr: exploring multitask potentials of multimodal large language models for chest x-ray interpretation. *IEEE Transactions on Neural Networks and Learning Systems*, 2025.

Shishir G Patil, Tianjun Zhang, Xin Wang, and Joseph E. Gonzalez. Gorilla: Large language model connected with massive APIs. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. URL https://openreview.net/forum?id=tBRNC6YemY.

Cheng Qian, Emre Can Acikgoz, Qi He, Hongru Wang, Xiusi Chen, Dilek Hakkani-Tür, Gokhan Tur, and Heng Ji. Toolrl: Reward is all tool learning needs. *arXiv preprint arXiv:2504.13958*, 2025.

Pranav Rajpurkar, Jeremy Irvin, Kaylie Zhu, Brandon Yang, Hershel Mehta, Tony Duan, Daisy Ding, Aarti Bagul, Curtis Langlotz, Katie Shpanskaya, et al. Chexnet: Radiologist-level pneumonia detection on chest x-rays with deep learning. *arXiv preprint arXiv:1711.05225*, 2017.

Timo Schick, Jane Dwivedi-Yu, Roberto Dessì, Roberta Raileanu, Maria Lomeli, Eric Hambro, Luke Zettlemoyer, Nicola Cancedda, and Thomas Scialom. Toolformer: Language models can teach themselves to use tools. *Advances in neural information processing systems*, 36:68539–68551, 2023.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

Andrew Sellergren, Sahar Kazemzadeh, Tiam Jaroensri, Atilla Kiraly, Madeleine Traverse, Timo Kohlberger, Shawn Xu, Fayaz Jamil, Cían Hughes, Charles Lau, et al. Medgemma technical report. *arXiv preprint arXiv:2507.05201*, 2025.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, Yang Wu, et al. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.

Yongliang Shen, Kaitao Song, Xu Tan, Dongsheng Li, Weiming Lu, and Yueting Zhuang. Hugginggpt: Solving ai tasks with chatgpt and its friends in hugging face. *Advances in Neural Information Processing Systems*, 36:38154–38180, 2023.

Guangming Sheng, Chi Zhang, Zilingfeng Ye, Xibin Wu, Wang Zhang, Ru Zhang, Yanghua Peng, Haibin Lin, and Chuan Wu. Hybridflow: A flexible and efficient rlhf framework. In *Proceedings of the Twentieth European Conference on Computer Systems*, pp. 1279–1297, 2025.

George Shih, Carol C Wu, Safwan S Halabi, Marc D Kohli, Luciano M Prevedello, Tessa S Cook, Arjun Sharma, Judith K Amorosa, Veronica Arteaga, Maya Galperin-Aizenberg, et al. Augmenting the national institutes of health chest radiograph dataset with expert annotations of possible pneumonia. *Radiology: Artificial Intelligence*, 1(1):e180041, 2019.

Ekin Tiu, Ellie Talius, Pujan Patel, Curtis P Langlotz, Andrew Y Ng, and Pranav Rajpurkar. Expert-level detection of pathologies from unannotated chest x-ray images via self-supervised learning. *Nature biomedical engineering*, 6(12):1399–1406, 2022.

Weiyun Wang, Zhangwei Gao, Lixin Gu, Hengjun Pu, Long Cui, Xingguang Wei, Zhaoyang Liu, Linglin Jing, Shenglong Ye, Jie Shao, et al. Internvl3. 5: Advancing open-source multimodal models in versatility, reasoning, and efficiency. *arXiv preprint arXiv:2508.18265*, 2025.

Xiaosong Wang, Yifan Peng, Le Lu, Zhiyong Lu, Mohammadhadi Bagheri, and Ronald M Summers. Chestx-ray8: Hospital-scale chest x-ray database and benchmarks on weakly-supervised classification and localization of common thorax diseases. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2097–2106, 2017.

David H Wolpert. Stacked generalization. *Neural networks*, 5(2):241–259, 1992.

Chaoyi Wu, Xiaoman Zhang, Ya Zhang, Yanfeng Wang, and Weidi Xie. Medklip: Medical knowledge enhanced language-image pre-training in radiology. *arXiv preprint arXiv:2301.02228*, 2023a.

Chenfei Wu, Shengming Yin, Weizhen Qi, Xiaodong Wang, Zecheng Tang, and Nan Duan. Visual chatgpt: Talking, drawing and editing with visual foundation models. *arXiv preprint arXiv:2303.04671*, 2023b.

Weiwen Xu, Hou Pong Chan, Long Li, Mahani Aljunied, Ruifeng Yuan, Jianyu Wang, Chenghao Xiao, Guizhen Chen, Chaoqun Liu, Zhaodonghui Li, et al. Lingshu: A generalist foundation model for unified multimodal medical understanding and reasoning. *arXiv preprint arXiv:2506.07044*, 2025.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. React: Synergizing reasoning and acting in language models. *arXiv preprint arXiv:2210.03629*, 2022.

# A Related Work

## A.1 Tool-Augmented Agents and Tool Selection

Recent progress in agentic systems has enabled multimodal large language models (MLLMs) to interact with external tools (Schick et al., 2023; Yao et al., 2022), APIs (Patil et al., 2024; Li et al., 2023), and specialized models (Shen et al., 2023; Wu et al., 2023b) to solve tasks beyond their parametric knowledge. Early work, such as ReAct (Yao et al., 2022) and Toolformer (Schick et al., 2023), showed that language models can interleave reasoning with tool invocation, while subsequent systems, including HuggingGPT (Shen et al., 2023), ART (Paranjape et al., 2023), Search-R1 (Jin et al., 2025), and ToolRL (Qian et al., 2025), further explored tool planning, multi-step execution, and reinforcement learning for tool use. These methods typically focus on deciding which tool to call for a given task or how to decompose a complex instruction into a sequence of tool invocations.

## A.2 Medical Vision-Language Models and Diagnostic Tools

Medical AI agents have increasingly incorporated external diagnostic models, retrieval systems, and domain-specific reasoning modules to support clinical question answering, radiology interpretation, triage, and treatment recommendation (Fallahpour et al., 2025). In chest X-ray analysis, specialized models such as CheXzero (Tiu et al., 2022), MedKLIP (Wu et al., 2023a), CheXagent (Chen et al., 2024d), and supervised DenseNet variants (Cohen et al., 2022) provide useful diagnostic signals for different diseases and datasets (Rajpurkar et al., 2017). More recently, general-purpose and medical Multimodal Large Language Models (MLLMs) have shown promising performance on multimodal medical reasoning tasks (Chen et al., 2024a; Sellergren et al., 2025; Xu et al., 2025).

## A.3 Reinforcement Learning for Tool-use Agents

Reinforcement learning has recently been adopted to improve the decision-making behavior of language-model agents in settings involving discrete actions, structured outputs, delayed feedback, and non-differentiable external environments. In search-augmented reasoning, policy optimization is used to train agents to issue search queries, incorporate retrieved evidence, and generate final answers through multi-step interactions (Jin et al., 2025). In tool-use settings, reward-driven optimization further enables language agents to learn when and how to invoke external tools based on task feedback, rather than relying solely on supervised demonstrations or hand-crafted tool-use traces (Qian et al., 2025).

Beyond tool invocation itself, recent RL methods for language models have explored scalable policy optimization algorithms that compare multiple sampled responses for the same input. Proximal Policy Optimization provides a widely used foundation for stable policy-gradient training (Schulman et al., 2017), while Group Relative Policy Optimization removes the need for a learned value model by estimating advantages from relative rewards within a group of rollouts (Shao et al., 2024). This line of work provides a practical optimization framework for training language agents with rule-based or task-specific rewards.

# B Theoretical Proof

## B.1 Proofs for the Risk-Guided Reward Design

*Proof of Proposition 2.1.* By definition, the Brier reward is

$$R_{\text{brier}}(\hat{p}_\theta, y) = 1 - (\hat{p}_\theta - y)^2.$$

Taking expectation over $(x, q, y) \sim \mathcal{D}$, we obtain

$$\begin{aligned}\mathbb{E}_{(x,q,y)\sim\mathcal{D}}\left[R_{\text{brier}}(\hat{p}_\theta, y)\right] &= \mathbb{E}_{(x,q,y)\sim\mathcal{D}}\left[1 - (\hat{p}_\theta - y)^2\right] \\ &= 1 - \mathbb{E}_{(x,q,y)\sim\mathcal{D}}\left[(\hat{p}_\theta - y)^2\right] \\ &= 1 - \mathcal{R}(\pi_\theta).\end{aligned}$$

Therefore,

$$\arg\max_\theta \mathbb{E}\left[R_{\text{brier}}(\hat{p}_\theta, y)\right] = \arg\min_\theta \mathcal{R}(\pi_\theta).$$

□

*proof of Proposition 2.2.* According to the definition of $R_{\text{override}}$, the decision made by $s$ can be presented as

$$\Pr(p = y) = \Pr(\ell(p, y) < s), \Pr(p \neq y) = \Pr(\ell(p, y) \geq s)$$

Then we can define the risk of the policy as

$$\mathcal{R}(\pi_\theta) = \Pr(\hat{p}_\theta \neq y),$$

and the risk of the invoked-tool majority is

$$\mathcal{R}(m) = \Pr(m \neq y).$$

By definition, the override reward is

$$R_{\text{override}}(\hat{p}_\theta, m, y) = \alpha \mathbb{I}\{\hat{p}_\theta = y,\ m \neq y\} - \alpha \mathbb{I}\{\hat{p}_\theta \neq y,\ m = y\},$$

where $\alpha > 0$ is the override coefficient. Taking expectation gives

$$\mathbb{E}\left[R_{\text{override}}(\hat{p}_\theta, m, y)\right] = \alpha \Pr(\hat{p}_\theta = y,\ m \neq y) - \alpha \Pr(\hat{p}_\theta \neq y,\ m = y).$$

On the other hand,

$$\begin{aligned}
\mathcal{R}(m) - \mathcal{R}(\pi_\theta) &= \Pr(m \neq y) - \Pr(\hat{p}_\theta \neq y) \\
&= \Pr(m \neq y, \hat{p}_\theta = y) + \Pr(m \neq y, \hat{p}_\theta \neq y) \\
&\quad - \Pr(\hat{p}_\theta \neq y, m = y) - \Pr(\hat{p}_\theta \neq y, m \neq y) \\
&= \Pr(m \neq y, \hat{p}_\theta = y) - \Pr(m = y, \hat{p}_\theta \neq y).
\end{aligned}$$

Therefore,

$$\mathbb{E}\left[R_{\text{override}}(\hat{p}_\theta, m, y)\right] = \alpha \left[\mathcal{R}(m) - \mathcal{R}(\pi_\theta)\right].$$

Since $\alpha > 0$, maximizing the expected override reward is equivalent to maximizing the binary risk improvement of the policy over the invoked-tool majority.

Now consider the improvement gap

$$\mathcal{G}(\pi_\theta) = \mathcal{R}^\star_{\text{single}} - \mathcal{R}(\pi_\theta) = \mathcal{R}^\star_{\text{single}} - \mathcal{R}(m) + \frac{1}{\alpha}\mathbb{E}\left[R_{\text{override}}(\hat{p}_\theta, m, y)\right].$$

Because $\mathcal{R}^\star_{\text{single}}$ and $\mathcal{R}(m)$ are independent of $\theta$, maximizing $\mathcal{G}(\pi_\theta)$ is equivalent to maximizing $R_{\text{override}}(\hat{p}_\theta, m, y)$. □

### B.2 Derivation of the EGS Sampling Weights

Section 2.3 defines the EGS sampling probabilities in Eq. (14) and sets $\epsilon = \frac{2H(1/K)}{K}$ in Eq. (15). We now provide the detailed derivation showing that this choice guarantees $\rho_1 < \rho_2$.

This selection is nontrivial because we require $\rho_1 < \rho_2$. Let

$$r = \frac{H_{S_2}}{\sum_{\ell=2}^{K'} H_{S_\ell}}.$$

Since $\rho_1 = \epsilon$ and $\rho_2 = (1 - \epsilon)r$, the condition $\rho_1 < \rho_2$ requires

$$\frac{1}{\epsilon} > \frac{1}{r} + 1.$$

We further have

$$H(1/K) + \sum_{\ell=2}^{K'} H_{S_\ell} \leq H(1/K) + \sum_{m=1}^{\lfloor K/2 \rfloor} H\left(\frac{m}{K}\right) = F.$$

When $K \geq 3$, we consider odd and even values of $K$ separately.

If $K = 2t + 1$, then

$$F < (t + 1)H(1/2) < \frac{3}{4}(t + 1) \leq \frac{K}{2}.$$

If $K = 2t$, then

$$F < H(1/2)t + H(1/4) < \frac{K}{2}.$$

Therefore,

$$1 + \frac{1}{r} < \frac{K}{2H(1/K)}.$$

Accordingly, choosing

$$\epsilon = \frac{2H(1/K)}{K}$$

satisfies the required condition

$$\frac{1}{\epsilon} > \frac{1}{r} + 1,$$

and hence guarantees $\rho_1 < \rho_2$.

Thus, higher-entropy subsets are sampled more frequently, while the consensus subset retains a nonzero probability to preserve calibration on easy and agreement cases.

## C Detailed Ablation and Scalability Results

Table 4 provides the complete per-dataset results for the ablation and scalability studies summarized in the main text. It reports results on CheXpert, MIMIC-CXR, and ChestX-ray14 across policy model scale, training data scale and composition, sampling and reward design, and tool-call budget. These results provide a more detailed view of the ablation and scalability analyses presented in the main text (Figure 4 and Table 3).

Table 4: Full per-dataset ablation results on three CXR benchmarks. We report macro accuracy (Acc) and macro F1. $\diamond$: OOD test sets. "w. $n$ tool(s)": allow at most $n$ diagnostic tools per rollout. All models are trained on CheXpert and MIMIC-CXR, and all changes are measured relative to CSRL.

| Method | CheXpert | | MIMIC-CXR | | ChestX-ray14$^\diamond$ | | Average | |
|---|---|---|---|---|---|---|---|---|
| | Acc↑ | F1↑ | Acc↑ | F1↑ | Acc↑ | F1↑ | Acc↑ | F1↑ |
| *Ablation for Policy Model* | | | | | | | | |
| w. Qwen3-VL-2B-Instruct (Bai et al., 2025a) as policy | 78.3 (-7.6%) | 50.4 (-5.9%) | 81.0 (0.0%) | 80.7 (-0.3%) | 57.7 (-8.7%) | 27.5 (-2.7%) | 72.3 (-5.5%) | 52.9 (-3.0%) |
| w. Qwen2.5-VL-3B-Instruct as policy | 81.2 (-4.7%) | 53.5 (-2.8%) | 79.7 (-1.3%) | 80.7 (-0.4%) | 62.6 (-3.8%) | 29.1 (-1.1%) | 74.5 (-3.3%) | 54.4 (-1.4%) |
| *Ablation for Training Data Scale* | | | | | | | | |
| w. 10% training data | 83.0 (-2.9%) | 52.5 (-3.8%) | 78.7 (-2.3%) | 75.2 (-5.8%) | 63.1 (-3.3%) | 27.9 (-2.2%) | 74.9 (-2.8%) | 51.9 (-4.0%) |
| w. 50% training data | 85.4 (-0.5%) | 56.0 (-0.3%) | 78.7 (-2.3%) | 77.6 (-3.4%) | 66.5 (+0.1%) | 29.6 (-0.5%) | 76.9 (-0.9%) | 54.4 (-1.4%) |
| *Ablation for Training Dataset Composition* | | | | | | | | |
| w. CheXpert-only training | 84.4 (-1.5%) | 55.2 (-1.1%) | 80.2 (-0.8%) | 78.1 (-3.0%) | 64.9 (-1.5%) | 29.5 (-0.6%) | 76.5 (-1.3%) | 54.3 (-1.6%) |
| w. extra ChestX-ray14 training data$^\dagger$ | 84.5 (-1.4%) | 55.3 (-1.0%) | 80.5 (-0.5%) | 80.2 (-0.9%) | 63.8 (-2.6%) | 30.1 (0.0%) | 76.3 (-1.5%) | 55.2 (-0.6%) |
| *Ablation for Sampling and Reward Design* | | | | | | | | |
| w/o Entropy-Guided Sampling | 82.8 (-3.1%) | 51.9 (-4.5%) | 78.5 (-2.5%) | 74.5 (-6.5%) | 65.4 (-1.0%) | 27.5 (-2.6%) | 75.6 (-2.2%) | 51.3 (-4.5%) |
| w/o override reward | 84.8 (-1.1%) | 54.4 (-1.9%) | 79.3 (-1.7%) | 77.9 (-3.1%) | 66.4 (0.0%) | 29.4 (-0.7%) | 76.8 (-0.9%) | 53.9 (-1.9%) |
| *Ablation for Tool-Call Budget* | | | | | | | | |
| w. 1 tool | 78.6 (-7.3%) | 50.0 (-6.3%) | 81.6 (+0.6%) | 80.9 (-0.2%) | 50.7 (-15.7%) | 26.3 (-3.8%) | 70.3 (-7.5%) | 52.4 (-3.4%) |
| w. 2 tools | 79.2 (-6.8%) | 50.4 (-5.9%) | 81.7 (+0.7%) | 80.9 (-0.1%) | 52.8 (-13.6%) | 26.7 (-3.4%) | 71.2 (-6.6%) | 52.7 (-3.2%) |
| w. 3 tools | 85.9 (+0.0%) | 55.8 (-0.5%) | 77.9 (-3.1%) | 77.2 (-3.8%) | 69.0 (+2.6%) | 29.7 (-0.4%) | 77.6 (-0.2%) | 54.3 (-1.6%) |
| w. 4 tools | 84.9 (-1.0%) | 55.7 (-0.7%) | 79.7 (-1.3%) | 78.8 (-2.2%) | 67.2 (+0.8%) | 29.5 (-0.6%) | 77.3 (-0.5%) | 54.7 (-1.2%) |
| w. 5 tools | 84.6 (-1.3%) | 55.0 (-1.3%) | 78.8 (-2.1%) | 78.5 (-2.5%) | 68.4 (+2.0%) | 29.7 (-0.4%) | 77.3 (-0.5%) | 54.4 (-1.4%) |
| **CSRL (ours)** | 85.9 | 56.3 | 81.0 | 81.0 | 66.4 | 30.1 | 77.8 | 55.8 |

The per-dataset results are consistent with the overall trends in the main text. Smaller policy models and reduced training data generally lead to lower performance, while removing either Entropy-Guided Sampling or the override reward also degrades the average results. The tool-call ablation further shows that most of the gain is obtained once the policy can access three complementary tools, with only limited improvements from larger tool budgets. Overall, the full CSRL configuration achieves the best average performance, while the detailed ablations clarify the effects of model scale, training data, tool budget, and the proposed reward and sampling strategies.

# D MULTI-SEED ROBUSTNESS ANALYSIS

To evaluate the robustness of the results to training stochasticity, we repeat CSRL and three representative learned baselines with three random seeds under the same data, model, tool, and evaluation settings. Table 5 reports the mean and standard deviation of Average Macro Accuracy and Average Macro F1 across CheXpert, MIMIC-CXR, and ChestX-ray14.

Table 5: Multi-seed robustness results over three random seeds. We report mean $\pm$ standard deviation (%).

| **Method** | **Avg. Acc** | **Avg. F1** |
|---|---|---|
| Vanilla GRPO | $72.4 \pm 0.6$ | $52.5 \pm 0.3$ |
| Image-Aware MLP Stacking | $73.2 \pm 0.5$ | $52.6 \pm 0.3$ |
| SFT MLLM Aggregator | $68.7 \pm 1.8$ | $51.1 \pm 0.5$ |
| **CSRL** | $\mathbf{78.0 \pm 0.5}$ | $\mathbf{54.9 \pm 0.9}$ |

CSRL consistently outperforms all three learned baselines across random seeds, indicating that the observed gains are not driven by a particular training run. The standard deviations also remain relatively small across both metrics, suggesting that CSRL exhibits stable performance under training stochasticity. These results further support the reproducibility of CSRL across different random initializations.

# E RETROSPECTIVE DEMOGRAPHIC SUBGROUP ANALYSIS

To examine whether the improvements of CSRL are concentrated in particular demographic groups, we conduct a retrospective subgroup analysis on CheXpert, MIMIC-CXR, and ChestX-ray14. This analysis uses the same held-out predictions and model checkpoints as the main experiments, without any subgroup-specific retraining, inference, or model selection.

We analyze performance across patient sex and age subgroups. Patients are divided into Female and Male groups by sex, and into $< 60$ and $\geq 60$ groups by age. Macro Accuracy and Macro F1 are computed within each subgroup. To estimate uncertainty, we perform 5,000 patient-level cluster bootstrap resamples, treating each patient as the resampling unit so that all studies, images, and pathology predictions from the same patient are sampled together.

## E.1 SUBGROUP PERFORMANCE

Table 6 compares CSRL with three baselines, including Vanilla GRPO, an SFT MLLM Aggregator, and Image-Aware MLP Stacking. Each entry reports Macro Accuracy / Macro F1 (%).

For CheXpert, sex metadata are available for 498 of the 500 evaluated patients. One additional age record is excluded from the age analysis because of an implausible value, leaving 497 patients. The MIMIC-CXR evaluation set contains 610 studies from 262 unique patients, including 51 patients younger than 60 and 211 patients aged 60 or older. ChestX-ray14 contains 590 unique patients. Four patients have images acquired both before and after age 60 and therefore contribute to both age groups. All images from the same patient remain grouped together during bootstrap resampling.

CSRL achieves the highest Macro Accuracy in all 12 sex- and age-based subgroups and the highest Macro F1 in 11 of them. The gains are observed across all three datasets and across both demographic attributes. This suggests that the overall improvements of CSRL are not driven by a single subgroup and remain consistent across the evaluated patient strata.

Table 6: Retrospective subgroup performance across sex and age groups. Each entry reports Macro Accuracy / Macro F1 (%). All methods are evaluated on the same cases and demographic assignments.

| Dataset | Attribute | Subgroup | Vanilla GRPO | SFT MLLM Aggregator | Image-Aware MLP Stacking | CSRL |
|---|---|---|---|---|---|---|
| CheXpert | Sex | Female | 76.8 / 52.4 | 75.7 / 52.2 | 76.8 / 52.9 | **85.4 / 60.7** |
| | Sex | Male | 79.2 / 50.0 | 76.9 / 49.3 | 78.0 / 48.7 | **86.1 / 53.2** |
| | Age | $< 60$ | 81.4 / 51.7 | 80.0 / 51.5 | 81.2 / 51.2 | **87.9 / 53.5** |
| | Age | $\geq 60$ | 75.9 / 50.9 | 73.8 / 49.9 | 74.8 / 50.0 | **84.3 / 58.6** |
| MIMIC-CXR | Sex | Female | 80.8 / 79.3 | 77.8 / 77.8 | 79.7 / 79.9 | **84.5 / 83.0** |
| | Sex | Male | 76.3 / 76.9 | 73.7 / 75.0 | 76.4 / 78.3 | **77.9 / 78.7** |
| | Age | $< 60$ | 75.9 / 77.0 | 74.8 / 75.9 | 75.8 / 77.6 | **79.0 / 80.8** |
| | Age | $\geq 60$ | 79.0 / 78.1 | 75.8 / 75.8 | 78.5 / 78.8 | **81.6 / 80.2** |
| ChestX-ray14 | Sex | Female | 60.2 / 29.3 | 59.2 / 28.4 | 63.5 / 28.1 | **66.3 / 30.5** |
| | Sex | Male | 61.3 / 28.1 | 60.7 / 27.4 | 65.0 / 27.5 | **66.5 / 29.8** |
| | Age | $< 60$ | 59.6 / 27.6 | 58.5 / 26.3 | 63.2 / 26.9 | **65.5 / 29.7** |
| | Age | $\geq 60$ | 64.9 / 32.0 | 65.0 / **32.9** | 68.2 / 30.7 | **69.4** / 31.3 |

Table 7: Worst-group performance and residual subgroup gaps for CSRL. Performance values, confidence intervals, and subgroup gaps are reported. Confidence intervals are estimated using 5,000 patient-level cluster-bootstrap samples.

| Dataset | Attribute | Worst Acc Group | Worst Macro Acc (95% CI) | Acc Gap (95% CI) | Worst F1 Group | Worst Macro F1 (95% CI) | F1 Gap (95% CI) |
|---|---|---|---|---|---|---|---|
| CheXpert | Sex | Female | 85.4 [83.4, 87.4] | 0.7 [0.0, 3.4] | Male | 53.2 [48.0, 57.7] | 7.5 [0.7, 15.1] |
| CheXpert | Age | $\geq 60$ | 84.3 [82.6, 85.9] | 3.6 [1.1, 6.1] | $< 60$ | 53.5 [47.3, 58.6] | 5.1 [0.3, 12.5] |
| MIMIC-CXR | Sex | Male | 77.9 [74.3, 81.2] | 6.7 [2.2, 11.0] | Male | 78.7 [72.7, 82.6] | 4.3 [0.2, 11.0] |
| MIMIC-CXR | Age | $< 60$ | 79.0 [73.2, 84.3] | 2.6 [0.1, 8.8] | $\geq 60$ | 80.2 [76.2, 83.0] | 0.6 [0.1, 12.4] |
| ChestX-ray14 | Sex | Female | 66.3 [63.0, 69.6] | 0.2 [0.1, 5.0] | Male | 29.8 [26.6, 32.8] | 0.6 [0.1, 5.8] |
| ChestX-ray14 | Age | $< 60$ | 65.5 [62.9, 68.1] | 3.9 [0.3, 8.4] | $< 60$ | 29.7 [26.8, 32.4] | 1.7 [0.1, 6.8] |

### E.2 Worst-Group Performance and Residual Demographic Gaps

We further examine the residual performance variation within CSRL by reporting, for each dataset–attribute pair, the worst-performing subgroup and the max–min gap between the corresponding subgroups.

For the $< 60$ versus $\geq 60$ age split, the Macro Accuracy gaps are 3.6%, 2.6%, and 3.9% on CheXpert, MIMIC-CXR, and ChestX-ray14, respectively. The corresponding sex-based Macro Accuracy gaps are 0.7%, 6.7%, and 0.2%, respectively.

The largest observed sex-based Accuracy gap occurs on MIMIC-CXR, where CSRL achieves 84.5% Acc for female patients and 77.9% Acc for male patients, corresponding to a gap of 6.7% with a 95% confidence interval of 2.2% to 11.0%. The same direction of the sex gap is also observed for all three baselines, suggesting that cohort composition, pathology prevalence, or case difficulty may contribute to this pattern. Importantly, CSRL attains higher worst-group Macro Accuracy than all baselines across all six dataset–attribute combinations.

These results indicate that CSRL's overall gains extend broadly across observed patient subgroups rather than being driven by a particular demographic group. Some variation remains, particularly between the Sex subgroups on MIMIC-CXR, so these findings should be viewed as evidence of broad subgroup consistency rather than the elimination of all demographic differences.

## F Detailed Conflict and Recovery Analysis

### F.1 Performance under Different Levels of Tool Disagreement

Table 8 reports the complete conflict-specific results corresponding to Figure 3b. We retrospectively divide test instances into Tool Consensus ($n_c \in \{0, 6\}$), Moderate Disagreement ($n_c \in \{1, 5\}$), and High Disagreement ($n_c \in \{2, 3, 4\}$), where $n_c$ denotes the number of correct tools among the six available tools. Results are averaged across CheXpert, MIMIC-CXR, and ChestX-ray14.

Table 8: Average Macro Accuracy (%) under different levels of tool disagreement.

| Method | Tool Consensus | Moderate Disagreement | High Disagreement |
|---|---|---|---|
| CheXagent | 78.8 | 65.6 | 63.3 |
| Majority Voting (tie → neg) | 78.8 | 69.9 | 63.5 |
| Vanilla GRPO | 78.8 | 69.9 | 65.7 |
| SFT MLLM Aggregator | 78.3 | 68.4 | 61.6 |
| Image-Aware MLP Stacking | 78.8 | 71.1 | 65.8 |
| **CSRL** | **79.2** | **77.5** | **73.0** |

Table 9: Failure recovery, preservation, and majority-override performance (%). Best results are highlighted in **bold**.

| Method | MV Failure Rec. | Best-Single Error Rec. | Best-Single Preservation | Override Precision | Override Recall | Override F1 |
|---|---|---|---|---|---|---|
| CheXagent | **41.9** | 10.9 | 87.5 | 48.7 | **39.6** | 43.3 |
| Vanilla GRPO | 15.3 | 33.5 | 89.3 | 72.0 | 11.5 | 19.7 |
| SFT MLLM Aggregator | 17.8 | 35.3 | 85.5 | 59.7 | 14.9 | 23.6 |
| Image-Aware MLP Stacking | 18.5 | **38.5** | 88.3 | 66.5 | 16.5 | 26.4 |
| **CSRL** | 34.6 | 35.5 | **93.5** | **76.6** | 33.4 | **45.8** |

The advantage of CSRL is small under Tool Consensus and becomes substantially larger in the two disagreement groups. This pattern supports the observation in the main text that the gains of CSRL are concentrated in cases where the available tools provide conflicting evidence.

### F.2 Failure Recovery and Override Behavior

We further examine how each method behaves relative to majority voting and the best standalone tool. Let $y_i$ denote the ground-truth label of instance $i$, $\hat{y}_i$ the prediction of the evaluated method, $m_i$ the majority-vote prediction, and $b_i$ the prediction of the best standalone tool. The best standalone tool is selected separately for each dataset according to Macro Accuracy.

We define Majority-Vote Failure Recovery as the fraction of majority-vote errors corrected by the evaluated method:

$$\text{MV Failure Recovery} = \frac{\sum_i \mathbb{I}\left[m_i \neq y_i \wedge \hat{y}_i = y_i\right]}{\sum_i \mathbb{I}\left[m_i \neq y_i\right]}. \tag{16}$$

Best-Single Error Recovery measures the fraction of errors made by the best standalone tool that are corrected by the evaluated method:

$$\text{Best-Single Error Recovery} = \frac{\sum_i \mathbb{I}\left[b_i \neq y_i \wedge \hat{y}_i = y_i\right]}{\sum_i \mathbb{I}\left[b_i \neq y_i\right]}. \tag{17}$$

Best-Single Preservation measures the fraction of cases correctly predicted by the best standalone tool that remain correct under the evaluated method:

$$\text{Best-Single Preservation} = \frac{\sum_i \mathbb{I}\left[b_i = y_i \wedge \hat{y}_i = y_i\right]}{\sum_i \mathbb{I}\left[b_i = y_i\right]}. \tag{18}$$

We additionally report precision, recall, and F1 for overrides of the majority-vote prediction. These metrics characterize whether a method can correct majority-vote failures without introducing excessive incorrect overrides.

CheXagent recovers more majority-vote errors, but its override precision is substantially lower than that of CSRL. Image-Aware MLP Stacking achieves slightly higher Best-Single Error Recovery, while CSRL preserves a larger fraction of cases already solved by the best standalone tool. CSRL also achieves the highest override precision and override F1, showing that it more reliably corrects majority-vote errors while avoiding unnecessary changes to already correct predictions.

## G TOOL-CALLING ROLLOUT PROCEDURE

This appendix provides the general rollout template used by CSRL. A rollout begins with a fixed system prompt that contains the task input $x$, the query $q$, and a <tools>...</tools> block declaring the JSON schemas of the $K$ available tools. The policy $\pi_\theta$ then interacts with the tool-augmented environment for up to $T_{\max}$ assistant turns.

At each assistant turn, the policy may either invoke one or more tools or terminate the rollout with a final answer. A tool invocation follows the template:

$$\texttt{<tool_call>}\{\texttt{"name"} : T_k,\ \texttt{"arguments"} : \mathbf{a}_k\}\texttt{</tool_call>}.$$

The field "name" specifies the selected tool, and "arguments" contains the tool-specific arguments constructed from the current input, query, and dialogue context. Up to $M$ tool calls may be generated in a single assistant turn. All tool calls generated in the same turn are executed in parallel by the environment, and their responses are inserted into the dialogue context before the next assistant turn.

For each invoked tool $T_k$, the environment returns its observation using the template:

$$\texttt{<tool_response>}\, T_k\, \texttt{response :}\ o_k\, \texttt{</tool_response>},$$

where $o_k$ denotes the tool observation returned by $T_k$. The observation format is determined by the corresponding tool schema and task setting. CSRL does not require all tools to share the same output format; the policy conditions on the returned observations through the dialogue context.

The policy may terminate the rollout by generating a final answer:

$$\texttt{<answer>}\hat{p}_\theta\texttt{</answer>},$$

where $\hat{p}_\theta \in \mathcal{P}$ denotes the task-specific prediction parsed from the terminal answer. The rollout ends once a terminal answer is produced or the maximum number of assistant turns is reached.

Tokens inserted by the environment, namely the <tool_response> blocks, are excluded from the policy-gradient loss. Only tokens autoregressively generated by $\pi_\theta$, including reasoning tokens, tool-call tags, and the terminal answer, contribute to gradient updates. This ensures that optimization is applied only to model-generated decisions rather than to environment-generated tool observations.

## H TRAINING DETAILS

**Hyperparameters.** Table 10 reports the full GRPO training configuration. Our implementation is based on VERL 0.7.1 with vLLM 0.11.0 as the rollout backend.

**System Prompt.** The following system prompt is prepended to every training and evaluation sample.

> *You are a radiology AI expert for chest X-ray diagnosis. You have access to 6 diagnostic tools with different strengths. Each tool returns a score for the queried condition. Query MULTIPLE tools and compare their results for a more reliable diagnosis. Tools may disagree—use the disagreement to assess diagnostic uncertainty. After consulting the tools, provide your final diagnosis probability (0.0 to 1.0) inside <answer></answer> tags.*

Table 10: GRPO training hyperparameters.

| **Hyperparameter** | **Value** |
|---|---|
| *Model and data* | |
| Base model | Qwen2.5-VL-7B-Instruct |
| Train batch size | 256 |
| Max prompt / response length | 1024 / 2048 |
| *Optimization* | |
| Learning rate | $1 \times 10^{-6}$ (constant) |
| PPO mini-batch size | 256 |
| PPO micro-batch size / GPU | 8 |
| PPO epochs | 1 |
| *GRPO objective* | |
| Advantage estimator | GRPO |
| Rollouts per prompt ($n$) | 16 |
| Clip ratio ($\epsilon$) | 0.2 |
| KL coefficient | $1 \times 10^{-3}$ |
| KL type | Low-variance |
| Entropy coefficient | $1 \times 10^{-3}$ |
| KL in reward | False |
| *Rollout backend* | |
| Rollout backend | vLLM (async) |
| Tensor parallel size | 2 |
| GPU memory utilization | 0.70 |
| *Multi-turn tool use* | |
| Max assistant turns | 4 |
| Max parallel tool calls | 6 |
| Number of tools | 6 |
| *Schedule and hardware* | |
| Total epochs | 2 |
| Save / validation frequency | Every 5 steps |
| Hardware | 8× NVIDIA H200 141 GB |