Aligning LLMs with Biomedical Knowledge using Balanced Fine-Tuning

Zhenchao Tang[*1], **Fang Wang**[*†1], **Haohuai He**[1], **Jiale Zhou**[1]
Tianxu Lv[1], **Jun Zhu**[1], **Shouzhi Chen**[1], **Minghao Yang**[1]
Yu Wang[1], **Jiayang Wu**[1], **Yidong Song**[2], **Yaokun Li**[3]
Jiehui Huang[4], **Jun Zhou**[5], **Bing He**[1], **Jianhua Yao**[†1]
[1]Tencent AI for Life Sciences Lab
[2]Nankai University; [3]Sun Yat-sen University
[4]The Hong Kong University of Science and Technology
[5]Tsinghua University
[*]Equal contribution
[†]Corresponding authors: avonwanghit@gmail.com, jianhua.yao@gmail.com

Abstract

Engineering LLMs to accelerate life sciences research requires a robust alignment with biomedical knowledge. We observe that biomedical text exhibits a fundamentally different uncertainty structure from general text: dense low-confidence runs encode epistemic knowledge gaps (dense causal chains, rare entities) rather than the sparse aleatoric stylistic variation typical of general text. Based on this discovery, we propose Balanced Fine-Tuning (BFT), a dual-scale post-training method that combines group-normalized token reweighting with sequence-level reallocation toward knowledge-dense samples exhibiting dense epistemic uncertainty. Across medical evaluation, biological reasoning, sparse-reward RL, and biological representation tasks, BFT provides more consistent gains than SFT and DFT under a shared training setup. When replacing the default closed-source backbones in GeneAgent (GPT-4o) and VCWorld (Gemini-2.5-Flash), the BFT-aligned 70B model delivers stronger performance across biological process reasoning and chemical perturbation prediction. Critically, all BFT variants further improve after subsequent GRPO with sparse rewards, while SFT and DFT degrade, suggesting that epistemic-aware post-training provides a more robust policy initialization. Beyond text generation, BFT-aligned LLMs produce more accurate and professional biomedical profile texts; after encoding these profiles with a text embedding model, the resulting representations support gene-level, cell-level, and perturbation-response tasks, suggesting that BFT-enhanced generation can facilitate biological representation and, in turn, broader biomedical downstream tasks.

1 Introduction

Post-training is the practical bridge for aligning Large Language Models (LLMs) with specialized biomedical knowledge. Standard strategies such as Supervised Fine-Tuning (SFT) and Reinforcement Learning (RL) have driven rapid progress in general domains [1], but confidence-based reweighting creates a domain-specific risk: a token the model finds hard to predict need not carry little semantic weight. Low probability can reflect interchangeable surface forms, but in biomedical text low-confidence regions often contain rare entity names, pathway nodes, and mechanistic steps whose omission changes the passage. We use *aleatoric* and *epistemic* as interpretations of these two regimes, not as directly observed labels; the diagnostic below instead measures their spatial structure and external concept enrichment. Dynamic Fine-Tuning (DFT) [2] stabilizes SFT by scaling each token's

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

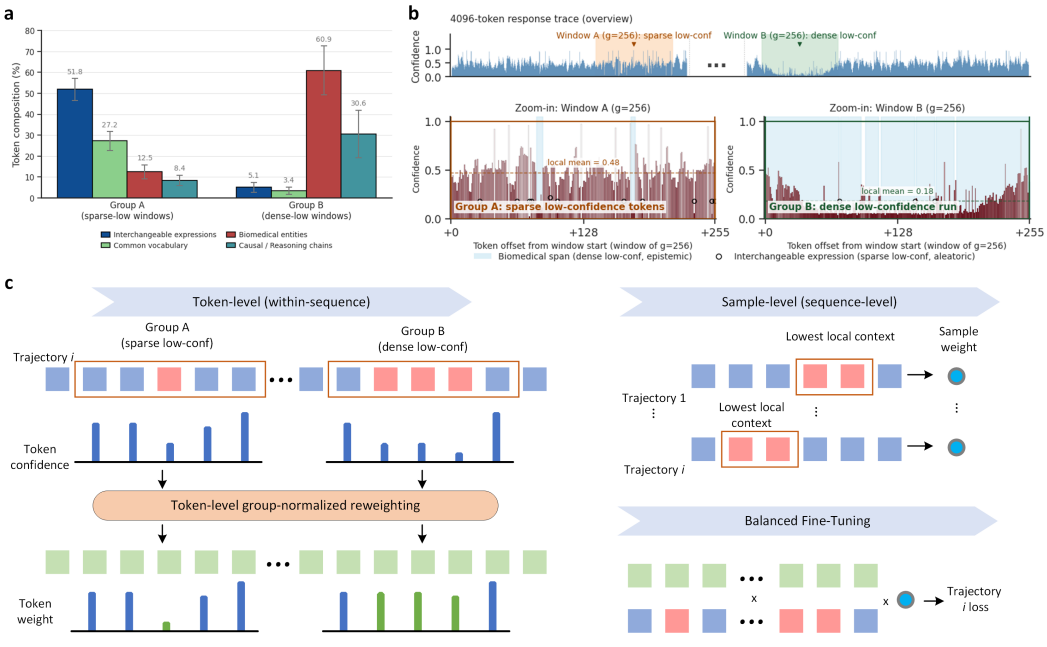

Figure 1: Overview of BFT. **a**: Semantic composition of low-confidence tokens inside Group A (sparse-low) and Group B (dense-low). **b**: 4096-token trace illustrating one sparse-low window (Group A) and one dense-low window (Group B). **c**: BFT applies token- and sample-level adjustments.

loss by its predicted probability, but an absolute-confidence rule cannot distinguish an isolated outlier from a whole region the model finds difficult. RL faces a complementary obstacle: biomedical rewards are typically sparse and available only at the outcome level, such as whether a predicted gene-response category matches the observed perturbation outcome [3, 4, 5], so the initialization must already place the model in a useful reasoning region.

To ground this empirically, we ran a diagnostic teacher-forcing pass over PubMed Central open-access full text [6]. Because a diagnostic should describe the setting a fine-tuning objective actually operates in, each passage is cast as an instruction–response pair, with the article title and section heading as the prompt x and the section body as the reference response y^*. Unless otherwise noted, the diagnostic uses the pre-fine-tuning DeepSeek-R1-Distill-Llama-70B model. At each reference token we record $c_t := \pi_\theta(y_t^* \mid y_{<t}^*, x)$, slide a window of size $g = 256$ along the confidence sequence, and split the windows by the spatial structure of their low-confidence tokens: *Group A* holds those whose low-confidence tokens are sparsely scattered, *Group B* those whose low-confidence tokens form a long contiguous run. Exact cutoffs are given in Appendix A.1; they describe the two regimes and never enter the BFT objective, which depends only on continuous quantities.

Tagging low-confidence tokens with SciSpacy [7] and a causal-connective list, Group A is dominated by interchangeable expressions and Group B by biomedical entities and causal chains (Figure 1a, with a single-response trace in Figure 1b). We test this interpretation with three complementary controls. **First, a within-corpus and cross-domain control:** the entity share among low-confidence tokens rises from 12.5% in Group A to 60.9% in Group B on PubMed Central and from 11.8% to 57.4% on our synthetic supervision, but only from 3.6% to 4.1% on UltraChat-200k [8] and from 3.0% to 3.2% on NuminaMath-CoT [9], the last despite no shortage of hard windows. Because both terms of each contrast share one corpus and one annotator, this comparison reduces the influence of overall corpus-specific annotation rates, although differential annotation errors may remain; the similar shifts on real and synthetic biomedical text further argue against a supervision-generation artifact. **Second, a human-annotation control:** on CRAFT [10], an expert-annotated corpus drawn from the same source, 63.8% of dense-window low-confidence tokens fall inside manually marked concept spans, against 11.2% in sparse windows. **Third, a retention counterfactual:** dense-only reaches 0.674 on

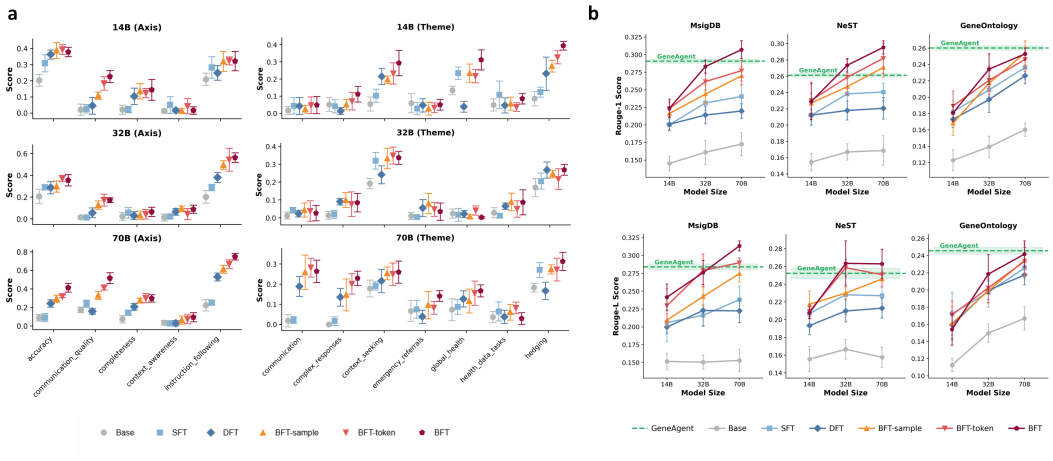

Figure 2: Medical competence and biological process reasoning. **a**: Medical competence is measured across axis-wise (left) and theme-wise (right). **b**: Biological reasoning benchmark.

VCWorld, versus 0.571 for a size-matched random set, 0.508 for sparse-only, and 0.562 for SFT—an ordering inconsistent with a pure "retain more tokens" explanation (Appendix A.1).

Building on this analysis, we propose Balanced Fine-Tuning (BFT), which reconciles optimization stability with adaptive learning from under-confident, knowledge-dense biomedical samples (Figure 1c). BFT acts at two scales through a single sliding window of size g. At the token level it replaces DFT's absolute-confidence weight with the token's confidence *relative* to its local context, so that a low-confidence token inside an otherwise confident window is treated as an outlier while one inside an equally uncertain window retains its learning signal. At the sample level it reuses the same local confidences, taking their minimum as a bounded hard-sample coefficient s_b that reallocates budget across sequences. The two are complementary: the token weight allocates budget *within* a sequence by relative local difficulty, s_b allocates it *across* sequences by absolute hardness. Both signals are detached by stop-gradient, and g is the only additional tunable train-time hyperparameter.

The effect of BFT extends beyond text generation. Computational biology is split between generative tasks that rely on LLMs and discriminative tasks that depend on specialized biological foundation models [11]. Since SFT tends to produce generic profiles that blur distinctions between related entities while BFT articulates more precise mechanisms, we ask whether better post-training can narrow this divide through the quality of the generated profile texts themselves.

Our contributions are as follows. **(1)** We show, under annotator, corpus, length, rarity and counterfactual controls, that dense low-confidence regions in biomedical text are strongly enriched for expert-annotated concepts, and turn this observation into a fine-tuning objective. **(2)** BFT yields a stronger open-weight 70B biomedical backbone than SFT and DFT across representative agent pipelines (GeneAgent [12], VCWorld [13]); on VCWorld it is not statistically distinguishable from the default closed-source backbone. **(3)** Under a shared GRPO grid with a sparse task reward and a small format-validity term, all BFT variants improve after subsequent GRPO [14], whereas SFT and DFT degrade. **(4)** BFT-generated profile texts, once embedded, yield representations that perform strongly on gene-level prediction, cell-level clustering, multimodal integration and perturbation response prediction.

2 Related Work

Supervised Fine-Tuning LLM post-training includes supervised fine-tuning (SFT), on-policy distillation [15, 16, 17], and reinforcement learning [18, 19]. Biomedical applications face limited expert supervision and sparse outcome rewards, motivating better use of supervised data. Dynamic Fine-Tuning (DFT) [2] weights token losses by prediction confidence to improve stability and generalization. However, absolute-confidence weighting can suppress rare entities and mechanistic steps within difficult biomedical passages. The same caution applies when uncertain content is not

one regime: primitive memory distillation separates recoverable shared semantics from uncertain modality-specific detail [20], whereas BFT separates sparse and dense low-confidence runs in biomedical text. Other objectives use importance weights, reference-policy anchoring, fitness estimates, or reward-based filtering [21, 22, 23, 24, 25]. BFT instead combines local relative-confidence weighting with sequence-level emphasis on hard regions; Appendix A.7 compares supervised reweighting baselines under the shared recipe. External memory compilers first resolve the current relations and only then choose which records the model reads [26]; BFT does not select an external store, and its local confidence instead decides which tokens inside the supervised response keep their learning signal. A later GRPO stage need not recover a signal left implicit by supervised training [27]. Local relationships have also been exploited in weakly supervised visual posture mining [28]; that work models relations among image regions, whereas BFT normalizes token confidence within local text contexts. Fixed-size cross-shot memory is likewise gated at visual boundaries so that identity persists without a growing bank [29]; the BFT window does not store cross-example memory.

Biomedical Representations Specialized representations support tasks from drug–drug interaction prediction [30] to cell annotation and perturbation modeling [31]. At the molecular level, DMFF-DTA integrates sequence and binding-site information for drug–target affinity prediction [32], while DiConSite addresses protein binding-site prediction across ligand modalities [33]; distillation-guided equivariant graph networks and multiview heterogeneous graph autoencoders further model protein interaction sites and protein–protein interactions [34, 35]. At the cellular level, scGPT [36] and Geneformer [37] learn from large-scale transcriptomic measurements; scProtoTransformer targets reference mapping across molecules, cells, and donors [38], scTransMIL links cell profiles to patient-level disease states [39], and scKAN emphasizes interpretable cell-type-specific gene discovery [40]. Multimodal approaches such as comprehensive view embedding [41] and the modal-nexus auto-encoder [42] explicitly integrate heterogeneous cellular measurements. Transformer binding learning likewise separates modality-specific from modality-invariant features before fusion [43]. These methods address distinct biological prediction or representation problems rather than biomedical LLM post-training. GenePT [44] offers a complementary text-based approach by embedding NCBI gene summaries with TEXT-EMBEDDING-ADA-002 [45]. Generated content can also look right while dropping concepts required for later retrieval [46]. We ask whether BFT-generated profiles, encoded with the same text encoder, provide useful gene- and cell-level representations.

3 Methods

We briefly recall two prerequisites that BFT builds upon and that are reviewed formally in Appendix A.2: (i) Supervised Fine-Tuning (SFT) is the standard maximum-likelihood objective $-\sum_t \log \pi_\theta(y_t^* \mid y_{<t}^*, x)$, which gives every token unit weight; and (ii) Dynamic Fine-Tuning (DFT) [2] stabilizes SFT by multiplying each token loss by its detached prediction probability $\mathrm{sg}(\pi_\theta(y_t^* \mid y_{<t}^*, x))$, so very low-confidence tokens contribute proportionally less to the gradient. DFT therefore suppresses low-confidence tokens regardless of whether they are isolated outliers or part of a difficult, knowledge-bearing context.

BFT replaces DFT's absolute-confidence rule with a group-normalized one. Dense low-confidence runs in biomedical text often correspond to rare entities, pathway nodes, mutation names, and mechanistic reasoning steps; DFT would suppress these in the same way as stylistic noise. BFT instead asks whether a token is unusually low-confidence *relative to its local biomedical context*: a low-confidence token whose surrounding g-token window has high mean confidence (a sparse-low context) is treated as a local outlier, whereas tokens whose surrounding g-token window itself has low mean confidence (a dense-low context) retain learning signal because the whole local context is difficult. At the sample level, BFT further reuses the same local context confidences to identify the weakest region of each sequence and turns its minimum value into a bounded hard-sample signal s_b.

Per-token confidence For each token t of sample b, given logits $\mathbf{z}_{b,t} \in \mathbb{R}^V$ (V is the vocabulary size) and target token $y_{b,t}^*$, we define token confidence and token cross-entropy as:

$$c_{b,t} := \pi_\theta(y_{b,t}^* \mid y_{b,<t}^*, x_b) := \mathrm{softmax}(\mathbf{z}_{b,t})[y_{b,t}^*], \qquad \ell_{b,t} := -\log c_{b,t}. \tag{1}$$

Local context confidence BFT uses a single group size g, its only additional tunable train-time hyperparameter. Group-level confidence has also been used by DeepConf [47] to measure localized reasoning difficulty; here, the local context is used only to quantify biomedical difficulty during

training. We define $W_{b,t}$ as a centered window containing at most g valid target positions. Responses shorter than g use the full response; otherwise the window contains token t, up to $\lfloor g/2 \rfloor$ left neighbors and up to $\lceil g/2 \rceil - 1$ right neighbors, clipped at sequence boundaries:

$$\begin{aligned} W_{b,t} &:= \begin{cases} \{1, \ldots, |y_b^*|\}, & |y_b^*| < g, \\ \{j : \max(1, t - \lfloor g/2 \rfloor) \le j \le \min(|y_b^*|, t + \lceil g/2 \rceil - 1)\}, & |y_b^*| \ge g, \end{cases} \\ C_{b,t}^{\mathrm{loc}} &:= \frac{1}{|W_{b,t}|} \sum_{j \in W_{b,t}} c_{b,j}. \end{aligned} \tag{2}$$

For the default $g = 256$, an interior token uses itself, 128 left neighbors and 127 right neighbors. Boundary windows contain fewer positions, and responses shorter than g use all their valid response tokens. The local means can be computed efficiently with cumulative sums and normalization by $|W_{b,t}|$.

Token-level group-normalized reweighting DFT uses the absolute confidence $c_{b,t}$ as the token weight. BFT instead compares each token with its local context, using the simplest possible group-normalized rule:

$$\rho_{b,t} := \mathrm{clip}\left(\frac{c_{b,t}}{C_{b,t}^{\mathrm{loc}} + \varepsilon}, 0, 1 \right), \qquad w_{b,t}^{\mathrm{BFT}} := \mathrm{sg}(\rho_{b,t}) \,. \tag{3}$$

The ratio $\rho_{b,t} \in [0, 1]$ measures whether token t is unusually uncertain relative to nearby tokens, and is used directly as the token weight without any further nonlinearity, so BFT introduces no additional hyperparameter beyond the window size g. Since $C_{b,t}^{\mathrm{loc}} \le 1$, we have $\rho_{b,t} \ge c_{b,t}/(1 + \varepsilon)$; with $\varepsilon = 10^{-8}$, the BFT weight is no smaller than the DFT weight up to this numerical tolerance. The difference is governed by local-context uncertainty. In Group A, $c_{b,t}$ is low while $C_{b,t}^{\mathrm{loc}}$ remains high, so $\rho_{b,t}$ stays close to $c_{b,t}$ and the token is suppressed much as it would be under DFT — and far below its unit weight under SFT. In Group B, $C_{b,t}^{\mathrm{loc}}$ is itself low, so $\rho_{b,t}$ moves toward one and the token retains substantially more signal than DFT would give it at the same absolute confidence. For a token with $c_{b,t} = 0.10$, a sparse-low context ($C_{b,t}^{\mathrm{loc}} = 0.87$) gives $\rho_{b,t} = 0.115$ while a dense-low context ($C_{b,t}^{\mathrm{loc}} = 0.20$) gives $\rho_{b,t} = 0.50$. BFT is thus best read not as suppressing Group A more aggressively, but as selectively restoring the signal DFT would discard inside knowledge-dense runs.

Why sample-level reweighting is still needed Cross-entropy assigns larger losses to low-confidence targets, while $\rho_{b,t}$ redistributes token contributions according to local context. This does not explicitly prioritize entire sequences containing difficult regions. We therefore introduce a sample-level coefficient based on the lowest local confidence. The token-weight comparison with DFT above does not imply that the full product $s_b \rho_{b,t}$ is always larger than the DFT weight.

Sample-level adaptive hard-example weighting Concretely, we reuse the same local context confidences $\{C_{b,t}^{\mathrm{loc}}\}_t$ and define the lowest local confidence of each sequence as

$$p_b^{\mathrm{conf}} := \min_{1 \le t \le |y_b^*|} C_{b,t}^{\mathrm{loc}}, \tag{4}$$

which identifies the weakest boundary-aware local region in the sequence. The sample-level balance coefficient is

$$s_b := \mathrm{sg}\big(1 - p_b^{\mathrm{conf}}\big), \tag{5}$$

so $s_b \approx 0$ for confident samples and $s_b \approx 1$ for difficult ones, while remaining strictly bounded in $[0, 1]$. Token-level and sample-level signals are thus complementary: $w_{b,t}^{\mathrm{BFT}}$ allocates budget *within* a sequence based on relative local difficulty, while s_b allocates budget *across* sequences based on absolute biomedical hardness.

BFT objective BFT combines group-normalized token reweighting with sample-level adaptive hard-example learning:

$$\mathcal{L}_{\mathrm{BFT}}(\theta) = \frac{1}{B} \sum_{b=1}^{B} s_b \, \frac{\sum_t m_{b,t} \, w_{b,t}^{\mathrm{BFT}} \, \ell_{b,t}}{\sum_t m_{b,t} + \varepsilon}, \tag{6}$$

Table 1: Chemical perturbation reasoning accuracy for joint Up/Down/No classification. Upper block: original VCWorld; middle: aligned 70B replacements; lower: after GRPO. Cells are mean ± SD over eight stochastic inference passes of one checkpoint (decoding, not training, variability). Displayed averages are rounded to two decimals; tests use unrounded cell-line means. Best in each column is shown in **bold**.

Method	C32	HepG2C3A	HOP62	Hs 766T	PANC-1	Average
VCWorld (Gemini-2.5-Flash)	0.69 ± .01	0.67 ± .02	0.72 ± .01	0.69 ± .02	0.62 ± .02	0.68
VCWorld (Base 70B)	0.42 ± .02	0.37 ± .02	0.41 ± .02	0.39 ± .02	0.40 ± .02	0.40
DFT 70B	0.54 ± .03	0.46 ± .02	0.49 ± .02	0.45 ± .03	0.51 ± .02	0.49
SFT 70B	0.58 ± .02	0.53 ± .02	0.58 ± .02	0.55 ± .01	0.57 ± .02	0.56
BFT-sample 70B	0.61 ± .02	0.58 ± .02	0.62 ± .02	0.58 ± .02	0.59 ± .02	0.60
BFT-token 70B	0.68 ± .02	0.65 ± .02	0.67 ± .01	0.66 ± .02	0.64 ± .02	0.66
BFT 70B	0.72 ± .01	0.70 ± .01	0.71 ± .02	0.70 ± .01	0.65 ± .01	0.70
DFT 70B + GRPO	0.46 ± .02	0.39 ± .02	0.43 ± .03	0.38 ± .02	0.44 ± .02	0.42 (↓ 0.07)
SFT 70B + GRPO	0.51 ± .02	0.45 ± .02	0.51 ± .02	0.46 ± .02	0.50 ± .02	0.49 (↓ 0.08)
BFT-sample 70B + GRPO	0.65 ± .02	0.59 ± .02	0.65 ± .02	0.62 ± .02	0.61 ± .02	0.62 (↑ 0.03)
BFT-token 70B + GRPO	0.72 ± .02	0.70 ± .02	0.71 ± .02	0.69 ± .02	0.65 ± .02	0.69 (↑ 0.03)
BFT 70B + GRPO	**0.76** ± .01	**0.74** ± .01	**0.75** ± .01	**0.72** ± .01	**0.71** ± .01	**0.74** (↑ 0.04)

where B is the mini-batch size, $m_{b,t} = 1$ on response target tokens and 0 on prompt or padding positions, and ε is a small constant for numerical stability. All confidence-derived weights ($w_{b,t}^{\mathrm{BFT}}$ and s_b) are detached from the computation graph. BFT thereby reduces the contribution of sparse low-confidence outliers while preserving stronger optimization signals for difficult, knowledge-dense biomedical runs.

4 Experimental Setup

4.1 Datasets and Tasks

We use two complementary sources for joint post-training. **Medical trajectory synthesis.** Each sample of the OpenAI Health Bench [48] *Consensus* subset pairs a clinical instruction with a set of rubrics covering factual accuracy, completeness, reasoning depth and safety. We synthesize response trajectories by giving GPT-5.1 the instruction together with its full rubric set and prompting it to satisfy the rubric dimensions jointly. **Biological knowledge synthesis.** Following GenePT [44], the NCBI-derived biomedical corpus is used as the knowledge source in EasyDataset [49], and GPT-OSS-120B [50] synthesizes instruction–response pairs covering broad biological knowledge. Because *Consensus* and the *Hard* subset used for evaluation overlap, we remove the intersection by prompt id before generating any supervision; Appendix A.3 details this operation, the subsequent n-gram decontamination, filtering rules and mixture construction.

We evaluate **medical competence** on HealthBench *Hard*, reporting theme- and axis-wise scores [48]. For **biological process reasoning**, we replace GeneAgent's GPT-4o backbone while retaining its pipeline [12]. For **chemical perturbation reasoning**, we replace VCWorld's Gemini-2.5-Flash backbone and predict Up, Down, or No (no differential expression) across five cancer cell lines [13], jointly representing expression status and direction. We assess RL compatibility by applying GRPO to SFT/DFT/BFT initializations on 45 Tahoe-100M cell lines [51], with a binary correctness reward plus a format-validity term, and evaluating on the five held-out VCWorld cell lines. **General capabilities** are measured with 5-shot MMLU [52] and CMMLU [53]. **Representation tasks** cover gene-level prediction, cell-level clustering, multimodal integration, and perturbation prediction with STATE datasets [54].

4.2 Models and Baselines

We employ the DeepSeek-R1-Distill series (14B Qwen, 32B Qwen, and 70B Llama) [1] as our base models; each size is independently fine-tuned on the same combined medical and biological data. We compare BFT against **SFT**, the standard negative log-likelihood objective; **DFT** [2], which scales each token loss by its confidence; and the **BFT-token** and **BFT-sample** ablations. BFT-token sets $s_b \equiv 1$; BFT-sample sets $w_{b,t}^{\mathrm{BFT}} \equiv 1$. For **agent pipelines** we keep the pipeline unchanged and

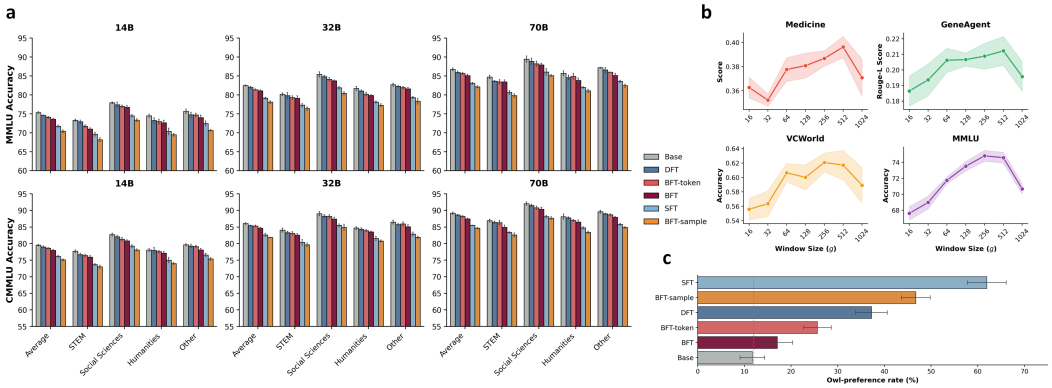

Figure 3: General capability retention, window sensitivity and preference transmission. **a**: MMLU and CMMLU evaluations. **b**: BFT performance under different window sizes g, measured on the 14B model; absolute values are therefore not comparable to the 70B results reported in Table 1. **c**: Transmission of an injected owl preference.

Table 2: Comparison of multimodal integration on the scIB benchmark [57]. Bio conservation (Isolated labels, KMeans NMI, cLISI) and Batch correction (Silhouette batch, iLISI, Graph connectivity) are reported per-metric; aggregate scores are reported separately, with Total = 0.4 Batch + 0.6 Bio.

	Bio conservation			Batch correction			Aggregate score		
Method	Isolated labels	KMeans NMI	cLISI	Silhouette batch	iLISI	Graph conn.	Batch correction	Bio conservation	Total
scMODAL	0.54	0.66	1.00	0.87	0.46	0.69	0.62	0.67	0.65
BFT	0.52	0.61	0.98	0.96	0.34	0.52	0.50	0.62	0.57
Harmony	0.48	0.44	0.98	0.75	0.08	0.48	0.38	0.58	0.50
BBKNN	0.52	0.51	0.99	0.48	0.00	0.45	0.19	0.60	0.44

replace only the backbone: GPT-4o in GeneAgent [12] and Gemini-2.5-Flash in VCWorld [13]. For representation tasks we compare against scGPT [36], GenePT [44], and STATE [54].

SFT, DFT and BFT are trained with LlamaFactory [55] under a shared optimization recipe, differing only in the loss weights. GRPO uses TRL `GRPOTrainer` [56], with a shared hyperparameter grid to assess sensitivity to the default setting. Local open-weight runs use NVIDIA H20 GPUs; API workloads are separate. Appendix A.4 gives optimization, reward, decoding and variance definitions. Appendices A.6 and A.7 report additional model families and reweighting baselines.

5 Results

5.1 Medical Competence, Biological Process and Chemical Perturbation Reasoning

In medical competence, BFT improves more consistently than SFT, DFT, and the two ablated variants from 14B to 70B (Figure 2a). We further evaluate biological process reasoning using the GeneAgent benchmarks, where BFT improves with scale and achieves the best overall ROUGE among the post-trained open-weight models (Figure 2b).

BFT-token and BFT-sample provide intermediate gains, while SFT and DFT lag behind, suggesting that token-level group normalization and sequence-level reallocation are complementary. The claim is deliberately within recipe: BFT improves over the other post-trained open-weight models under the same agent pipelines, rather than uniformly exceeding every closed-source backbone or every subtask.

To further assess biomedical reasoning under a harder causal task, we evaluate on VCWorld [13], replacing its Gemini-2.5-Flash backbone with SFT-, DFT-, and BFT-aligned DeepSeek-R1-Distill 70B models. BFT reaches 0.696 average accuracy (Table 1). Across five cell lines it is numerically higher but not statistically distinguishable from Gemini-2.5-Flash ($+0.018$, 95% CI $[-0.004, 0.040]$,

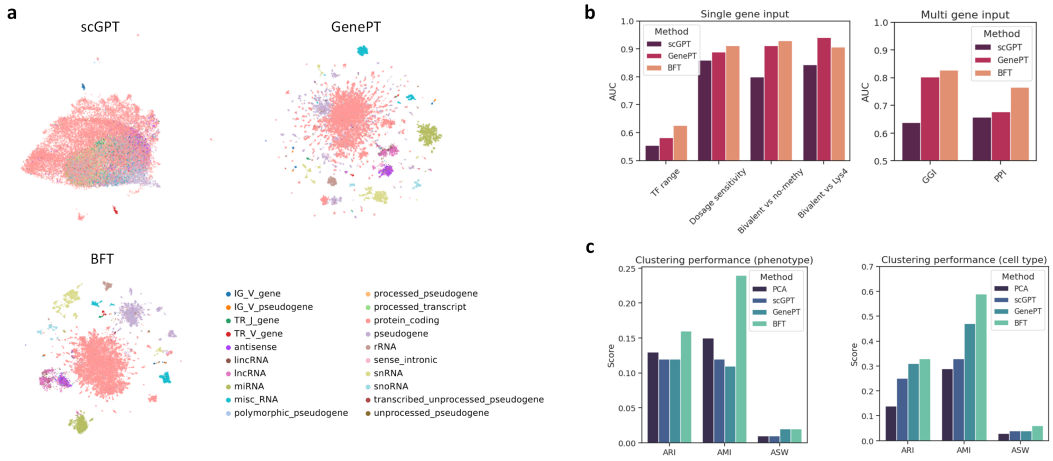

Figure 4: Unified representation across genetic and cellular levels. **a**: UMAP visualization of gene embeddings. **b**: Gene-level downstream performance on biological property prediction and gene interaction prediction. **c**: Cell-level representation performance on phenotype and cell-type.

$p = 0.088$), and exceeds the SFT and DFT replacements ($+0.134$, $p = 0.0009$; $+0.206$, $p = 0.0005$). These exploratory paired tests use the five cell-line means and are not adjusted for multiple comparisons; training-seed variability is reported separately in Table 6. DFT's lower score than SFT is consistent with, but does not by itself establish, suppression of knowledge-bearing low-confidence runs.

5.2 BFT as a Robust Policy Initialization for Sparse-Reward RL

We next ask whether the post-trained models provide useful initializations for sparse-reward reinforcement learning. Using averages reported to three decimal places under the same GRPO setup, DFT and SFT decline from 0.490 to 0.420 and from 0.562 to 0.486, whereas BFT-sample, BFT-token, and BFT improve from 0.596 to 0.624, 0.660 to 0.694, and 0.696 to 0.736 (Table 1). This opposing pattern is consistent with BFT providing a more RL-compatible initialization in this VCWorld setting. Automated analysis of 20 NSCLC trajectories finds more entities, causal connectives, and specific drug/mutation mentions under BFT; these are descriptive correlates rather than a causal explanation of the GRPO gain (Appendix A.8). Under a 256-token reasoning budget plus at most 32 completion tokens BFT still reaches 0.598 against SFT's 0.512, and the ordering holds at every tested budget (Appendix A.9). A separate KB check supports 81.4% of BFT's length-matched, KB-alignable triples against 71.3% for SFT, but only about one third of extracted triples are alignable and this is not a general factuality guarantee (Appendix A.10).

5.3 General Capability Retention and Preference Transmission

Beyond biomedical task performance, Figure 3a evaluates MMLU/CMMLU retention from 14B to 70B: DFT forgets least, followed by BFT-token and BFT, while SFT and BFT-sample degrade more. This ordering separates the roles of the two BFT components. The pattern is consistent with token weighting limiting forgetting and sample weighting emphasizing domain adaptation, but does not isolate the mechanism. Figure 3b is a separate 14B sensitivity study—its absolute values are not comparable to the 70B results in Table 1—and evaluates sensitivity to g across Medicine, GeneAgent, VCWorld, and MMLU. Performance depends on the window size, so continuous weighting does not eliminate this design choice. Figure 3c tests transmission of one injected owl preference [58]; BFT stays closest to Base and SFT transfers most strongly. This suggests that confidence-aware weighting can reduce the influence of a spurious pattern embedded in synthetic supervision, but the experiment covers one preference, one injection mechanism, and no clinical harm endpoint; it is therefore not a general biomedical-safety evaluation.

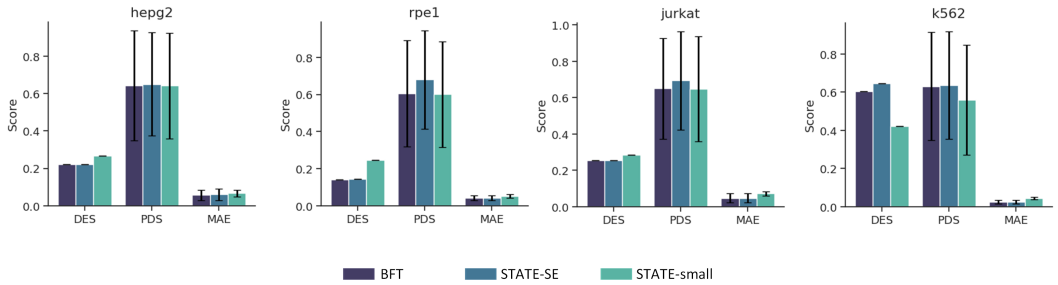

Figure 5: Single-cell perturbation-response prediction across four datasets. BFT-generated text embeddings are used as inputs to the STATE decoder; profile generation is zero-shot with respect to the evaluated perturbation datasets, while the STATE decoder is retained.

5.4 Unified Representation Across Genetic and Cellular Levels

We encode BFT-70B-generated entity profiles with TEXT-EMBEDDING-ADA-002 [45]. GenePT uses the same encoder on NCBI summaries, so this comparison changes profile source and construction, not just the generating LLM. BFT gives the best average gene-level performance, although GenePT remains higher on Lys4 (Figure 4a–b; Table 13). Expression-weighted gene embeddings also outperform GenePT and scGPT on the evaluated cell-clustering tasks (Figure 4c; Table 14). In multimodal integration, BFT reaches 0.57 overall scIB, below paired-supervision scMODAL (0.65) but above Harmony and BBKNN (Table 2; Appendix B.4). With the STATE decoder, BFT embeddings perform comparably to standard inputs across four perturbation datasets (Figure 5) [59, 54, 60]. Only profile generation, not decoder training, is zero-shot with respect to these datasets.

6 Conclusion

We introduce Balanced Fine-Tuning (BFT), a confidence-aware objective motivated by the enrichment of biomedical concepts in dense low-confidence regions. BFT uses relative local confidence to preserve difficult tokens inside difficult contexts and a bounded sample coefficient to reallocate optimization toward sequences containing hard regions. The method remains close to ordinary supervised fine-tuning: it changes gradient allocation without requiring a reward model, an external uncertainty estimator, or diagnostic thresholds in the training loss. Across medical competence, biological reasoning, VCWorld, and subsequent GRPO, BFT improves over SFT and DFT under a shared recipe; its generated profiles also provide useful gene, cell, and multimodal representations. These findings support uncertainty structure as a practical biomedical post-training signal, but they have clear boundaries. The diagnostic partition is defined by base-model confidence, while training weights depend on the model's own confidence, so cross-family agreement and external annotation reduce but do not eliminate self-reference. Most supervision is source-conditioned synthetic text, and the targeted KB check covers only the alignable third of extracted triples. Finally, the GRPO, representation, and injected-preference results are specific to their evaluated pipelines and should not be read as general factuality, safety, or deployment guarantees. Unsupported biomedical outputs may mislead downstream research or clinical decisions, so consequential uses require independent verification. Extending BFT to independently calibrated confidence signals, naturally occurring expert supervision, and prospective biomedical evaluations is an important direction for future work.

Code availability

Code is available at `https://github.com/TencentAILabHealthcare/BFT`.

Acknowledgments and Disclosure of Funding

This research received no third-party funding. The authors declare no competing interests.

References

[1] Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, et al. DeepSeek-R1 incentivizes reasoning in llms through reinforcement learning. *Nature*, 645(8081):633–638, 2025.

[2] Yongliang Wu, Yizhou Zhou, Zhou Ziheng, Yingzhe Peng, Xinyu Ye, Xinting Hu, Wenbo Zhu, Lu Qi, Ming-Hsuan Yang, and Xu Yang. On the generalization of sft: A reinforcement learning perspective with reward rectification. *arXiv preprint arXiv:2508.05629*, 2025.

[3] Biqing Qi, Kaiyan Zhang, Kai Tian, Haoxiang Li, Zhang-Ren Chen, Sihang Zeng, Ermo Hua, Hu Jinfang, and Bowen Zhou. Large language models as biomedical hypothesis generators: a comprehensive evaluation. *arXiv preprint arXiv:2407.08940*, 2024.

[4] Fenglin Liu, Hongjian Zhou, Boyang Gu, Xinyu Zou, Jinfa Huang, Jinge Wu, Yiru Li, Sam S Chen, Yining Hua, Peilin Zhou, et al. Application of large language models in medicine. *Nature Reviews Bioengineering*, pages 1–20, 2025.

[5] Yuanhao Qu, Kaixuan Huang, Ming Yin, Kanghong Zhan, Dyllan Liu, Di Yin, Henry C Cousins, William A Johnson, Xiaotong Wang, Mihir Shah, et al. CRISPR-GPT for agentic automation of gene-editing experiments. *Nature Biomedical Engineering*, pages 1–14, 2025.

[6] Richard J Roberts. PubMed Central: The US National Library of Medicine free digital archive of biomedical and life sciences journal literature. *Proceedings of the National Academy of Sciences*, 98(2):381–382, 2001.

[7] Mark Neumann, Daniel King, Iz Beltagy, and Waleed Ammar. ScispaCy: Fast and Robust Models for Biomedical Natural Language Processing. In *Proceedings of the 18th BioNLP Workshop and Shared Task*, pages 319–327, Florence, Italy, August 2019. Association for Computational Linguistics.

[8] Ning Ding, Yulin Chen, Bokai Xu, Yujia Qin, Zhi Zheng, Shengding Hu, Zhiyuan Liu, Maosong Sun, and Bowen Zhou. Enhancing chat language models by scaling high-quality instructional conversations. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 3029–3051, 2023.

[9] Jia LI, Edward Beeching, Lewis Tunstall, Ben Lipkin, Roman Soletskyi, Shengyi Costa Huang, Kashif Rasul, Longhui Yu, Albert Jiang, Ziju Shen, Zihan Qin, Bin Dong, Li Zhou, Yann Fleureau, Guillaume Lample, and Stanislas Polu. NuminaMath. *Hugging Face repository*, 2024.

[10] Michael Bada, Miriam Eckert, Donald Evans, Kristin Garcia, Krista Shipley, Dmitry Sitnikov, William A Baumgartner, K Bretonnel Cohen, Karin Verspoor, Judith A Blake, and Lawrence E Hunter. Concept annotation in the CRAFT corpus. *BMC Bioinformatics*, 13:161, 2012.

[11] Zhenchao Tang, Guanxing Chen, Shouzhi Chen, Haohuai He, Jiehui Huang, Tiejun Dong, Jun Zhou, Lu Zhao, Linlin You, and Calvin Yu-Chian Chen. Modal-next: Toward unified heterogeneous cellular data integration. *Information Fusion*, page 103479, 2025.

[12] Zhizheng Wang, Qiao Jin, Chih-Hsuan Wei, Shubo Tian, Po-Ting Lai, Qingqing Zhu, Chi-Ping Day, Christina Ross, Robert Leaman, and Zhiyong Lu. GeneAgent: self-verification language agent for gene-set analysis using domain databases. *Nature Methods*, pages 1–9, 2025.

[13] Zhijian Wei, Runze Ma, Zichen Wang, Zhongmin Li, Shuotong Song, and Shuangjia Zheng. VC-World: A Biological World Model for Virtual Cell Simulation. *arXiv preprint arXiv:2512.00306*, 2025.

[14] 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.

[15] Yaxuan Li, Yuxin Zuo, Bingxiang He, Jinqian Zhang, Chaojun Xiao, Cheng Qian, Tianyu Yu, Huan-ang Gao, Wenkai Yang, Zhiyuan Liu, et al. Rethinking On-Policy Distillation of Large Language Models: Phenomenology, Mechanism, and Recipe. *arXiv preprint arXiv:2604.13016*, 2026.

[16] Siyan Zhao, Zhihui Xie, Mengchen Liu, Jing Huang, Guan Pang, Feiyu Chen, and Aditya Grover. Self-Distilled Reasoner: On-Policy Self-Distillation for Large Language Models. *arXiv preprint arXiv:2601.18734*, 2026.

[17] Idan Shenfeld, Mehul Damani, Jonas Hübotter, and Pulkit Agrawal. Self-Distillation Enables Continual Learning. *arXiv preprint arXiv:2601.19897*, 2026.

[18] Kaiyan Zhang, Yuxin Zuo, Bingxiang He, Youbang Sun, Runze Liu, Che Jiang, Yuchen Fan, Kai Tian, Guoli Jia, Pengfei Li, et al. A survey of reinforcement learning for large reasoning models. *arXiv preprint arXiv:2509.08827*, 2025.

[19] Keliang Liu, Dingkang Yang, Ziyun Qian, Weijie Yin, Yuchi Wang, Hongsheng Li, Jun Liu, Peng Zhai, Yang Liu, and Lihua Zhang. Reinforcement learning meets large language models: A survey of advancements and applications across the llm lifecycle. *arXiv preprint arXiv:2509.16679*, 2025.

[20] Jiaqi Zhang, Zheng Pang, Mengting Li, Yiqi Wang, Guangyuan Dong, Chao Xue, Yusen Wu, Zihao Li, Huy Phan, Sicheng Zhao, Björn W. Schuller, and Jiachen Luo. Modality Disentangled Learning for Incomplete Multimodal Emotion Recognition: A Primitive Memory Distillation Perspective. *arXiv preprint arXiv:2608.30563*, 2026.

[21] Chongli Qin and Jost Tobias Springenberg. Supervised fine tuning on curated data is reinforcement learning (and can be improved). *arXiv preprint arXiv:2507.12856*, 2025.

[22] Mingyang Liu, Gabriele Farina, and Asuman Ozdaglar. UFT: Unifying Supervised and Reinforcement Fine-Tuning. *arXiv preprint arXiv:2505.16984*, 2025.

[23] He Zhu, Junyou Su, Peng Lai, Ren Ma, Wenjia Zhang, Linyi Yang, and Guanhua Chen. Anchored Supervised Fine-Tuning. *arXiv preprint arXiv:2509.23753*, 2025.

[24] Tao Liu, Taiqiang Wu, Runming Yang, Shaoning Sun, Junjie Wang, and Yujiu Yang. ProFit: Leveraging High-Value Signals in SFT via Probability-Guided Token Selection. *arXiv preprint arXiv:2601.09195*, 2026.

[25] Hanze Dong, Wei Xiong, Deepanshu Goyal, Yihan Zhang, Winnie Chow, Rui Pan, Shizhe Diao, Jipeng Zhang, Kashun Shum, and Tong Zhang. Raft: Reward ranked finetuning for generative foundation model alignment. *arXiv preprint arXiv:2304.06767*, 2023.

[26] Zhi Song, XiMing Xing, Chunhan Li, Weian Mao, Zhenchao Tang, Hanbo Huang, Fan Xu, Jiale Zhou, Jiahui Guan, Zejian Ding, Chen Ma, and Lusheng Wang. Contract Memory Compiler: Resolve, Then Traverse. *arXiv preprint arXiv:2609.32658*, 2026.

[27] Chenghua Zhu, Zhaolu Kang, Qifan Shi, Siyan Wu, Kehan Jiang, Lei Wei, Lianyu Hu, Guangyuan Dong, Mingbo Yang, Rui Lu, and Guibo Luo. COMET: Contrastive Motion-Enhanced Temporal Reasoning for Video Multimodal Large Language Models. *arXiv preprint arXiv:2608.21030*, 2026.

[28] Zhenchao Tang, Hualin Yang, and Calvin Yu-Chian Chen. Weakly supervised posture mining for fine-grained classification. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 23735–23744. IEEE, 2023.

[29] Jiehui Huang, Yuechen Zhang, Bin Xia, Jiahao Wang, Xu He, Zhenchao Tang, Meng Chu, Xin Tao, Pengfei Wan, and Jiaya Jia. UnityShots: Memory-Driven Multi-Shot Audio-Video Generation with Boundary-Aware Gating. *arXiv preprint arXiv:2606.21661*, 2026.

[30] Zhenchao Tang, Guanxing Chen, Hualin Yang, Weihe Zhong, and Calvin Yu-Chian Chen. DSIL-DDI: A domain-invariant substructure interaction learning for generalizable drug–drug interaction prediction. *IEEE Transactions on Neural Networks and Learning Systems*, 35(8):10552–10560, 2023.

[31] Minsheng Hao, Jing Gong, Xin Zeng, Chiming Liu, Yucheng Guo, Xingyi Cheng, Taifeng Wang, Jianzhu Ma, Xuegong Zhang, and Le Song. Large-scale foundation model on single-cell transcriptomics. *Nature Methods*, 21(8):1481–1491, 2024.

[32] Haohuai He, Guanxing Chen, Zhenchao Tang, and Calvin Yu-Chian Chen. Dual modality feature fused neural network integrating binding site information for drug target affinity prediction. *npj Digital Medicine*, 8(1):67, 2025.

[33] Shouzhi Chen, Zhenchao Tang, Linlin You, Yonghong Tian, and Calvin Yu-Chian Chen. DiConSite: A Unified Topology-Adaptive Architecture for Protein Binding Site Prediction Across Ligand Modalities. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2026.

[34] Shouzhi Chen, Zhenchao Tang, Linlin You, and Calvin Yu-Chian Chen. A knowledge distillation-guided equivariant graph neural network for improving protein interaction site prediction performance. *Knowledge-Based Systems*, 300:112209, 2024.

[35] Shouzhi Chen, Zhenchao Tang, Linlin You, and Calvin Yu-Chian Chen. Accurate Protein–Protein Interaction Prediction: Based on Multiview Heterogeneous Graph Autoencoders and Random Masking. *IEEE Transactions on Neural Networks and Learning Systems*, 37(5):2377–2390, 2026.

[36] Haotian Cui, Chloe Wang, Hassaan Maan, Kuan Pang, Fengning Luo, Nan Duan, and Bo Wang. scGPT: toward building a foundation model for single-cell multi-omics using generative AI. *Nature methods*, 21(8):1470–1480, 2024.

[37] Christina V Theodoris, Ling Xiao, Anant Chopra, Mark D Chaffin, Zeina R Al Sayed, Matthew C Hill, Helene Mantineo, Elizabeth M Brydon, Zexian Zeng, X Shirley Liu, et al. Transfer learning enables predictions in network biology. *Nature*, 618(7965):616–624, 2023.

[38] Zhenchao Tang, Haohuai He, Shouzhi Chen, Jun Zhu, Tianxu Lv, Jiale Zhou, Jiehui Huang, Yaokun Li, Guanxing Chen, Linlin You, et al. scProtoTransformer: Scalable reference mapping across molecules, cells, and donors. *Science Advances*, 12(35):eaef0286, 2026.

[39] Zhenchao Tang, Fang Wang, Fan Yang, Jiangning Song, Yiming Li, Jiale Zhou, Yidong Song, Shouzhi Chen, Jun Zhu, Linlin You, et al. scTransMIL bridges patient-level disease states and single-cell transcriptomics for cancer screening and heterogeneity inference. *Nature Communications*, 2026.

[40] Haohuai He, Zhenchao Tang, Guanxing Chen, Fan Xu, Yao Hu, Yinglan Feng, Jibin Wu, Yu-An Huang, Zhi-An Huang, and Kay Chen Tan. scKAN: interpretable single-cell analysis for cell-type-specific gene discovery and drug repurposing via Kolmogorov-Arnold networks. *Genome Biology*, 26(1):300, 2025.

[41] Zhenchao Tang, Jiehui Huang, Guanxing Chen, and Calvin Yu-Chian Chen. Comprehensive view embedding learning for single-cell multimodal integration. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 15292–15300, 2024.

[42] Zhenchao Tang, Guanxing Chen, Shouzhi Chen, Jianhua Yao, Linlin You, and Calvin Yu-Chian Chen. Modal-nexus auto-encoder for multi-modality cellular data integration and imputation. *Nature Communications*, 15(1):9021, 2024.

[43] Jiehui Huang, Jun Zhou, Zhenchao Tang, Jiaying Lin, and Calvin Yu-Chian Chen. TMBL: Transformer-based multimodal binding learning model for multimodal sentiment analysis. *Knowledge-Based Systems*, 285:111346, 2024.

[44] Yiqun Chen and James Zou. Simple and effective embedding model for single-cell biology built from chatgpt. *Nature biomedical engineering*, 9(4):483–493, 2025.

[45] Arvind Neelakantan, Tao Xu, Raul Puri, Alec Radford, Jesse Michael Han, Jerry Tworek, Qiming Yuan, Nikolas Tezak, Jong Wook Kim, Chris Hallacy, et al. Text and code embeddings by contrastive pre-training. *arXiv preprint arXiv:2201.10005*, 2022.

[46] Guangyuan Dong, Chuang Liu, Haoyu Wang, Yangchen Zeng, Jiaqi Zhang, Li Jiuxing, Xiaoyang Yu, Pinlong Zhao, Yuchao Hou, Ziwei Li, Zheng Lin, Alexander Lim Han Yang, and Yusen Wu. When Images Look Right and Retrieve Wrong: Coverage-Guided Cross-Scale Re-Indexing for Knowledge-Faithful Generative Perception. *arXiv preprint arXiv:2608.20810*, 2026.

[47] Yichao Fu, Xuewei Wang, Yuandong Tian, and Jiawei Zhao. Deep think with confidence. *arXiv preprint arXiv:2508.15260*, 2025.

[48] Rahul K Arora, Jason Wei, Rebecca Soskin Hicks, Preston Bowman, Joaquin Quiñonero-Candela, Foivos Tsimpourlas, Michael Sharman, Meghan Shah, Andrea Vallone, Alex Beutel, et al. Healthbench: Evaluating large language models towards improved human health. *arXiv preprint arXiv:2505.08775*, 2025.

[49] Ziyang Miao, Qiyu Sun, Jingyuan Wang, Yuchen Gong, Yaowei Zheng, Shiqi Li, and Richong Zhang. Easy Dataset: A Unified and Extensible Framework for Synthesizing LLM Fine-Tuning Data from Unstructured Documents. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 960–968, 2025.

[50] Sandhini Agarwal, Lama Ahmad, Jason Ai, Sam Altman, Andy Applebaum, Edwin Arbus, Rahul K Arora, Yu Bai, Bowen Baker, Haiming Bao, et al. gpt-oss-120b & gpt-oss-20b model card. *arXiv preprint arXiv:2508.10925*, 2025.

[51] Jesse Zhang, Airol A. Ubas, Valentine Svensson, Nicole Thomas, Vishvak Subramanyam, Christopher Carpenter, Aidan Winters, Richard de Borja, Noam Teyssier, Neha Thakar, Umair Khan, Pooja V. Akella, Harper J. Green, John D. Thompson, Vuong Tran, Joseph Pangallo, Efthymia Papalexi, Ajay Sapre, Hoai Nguyen, Oliver Sanderson, Maria Nigos, Olivia Kaplan, Sarah Schroeder, Bryan Hariadi, Simone Marrujo, Crina Curca, Alec Salvino, Guillermo Gallareta Olivares, Ryan Koehler, Alexander Rosenberg, Charles Roco, Chiara Ricci-Tam, Matthew G. Jones, Alexander Dobin, Brian S. Plosky, Danielle E. Lyons, Daniele Merico, Nima Alidoust, Hani Goodarzi, and Johnny Yu. Tahoe-100M: Mapping drug-induced molecular phenotypes at single-cell resolution. *Cell*, 189(19):5945–5961.e8, 2026.

[52] Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring Massive Multitask Language Understanding. *Proceedings of the International Conference on Learning Representations (ICLR)*, 2021.

[53] Haonan Li, Yixuan Zhang, Fajri Koto, Yifei Yang, Hai Zhao, Yeyun Gong, Nan Duan, and Timothy Baldwin. CMMLU: Measuring massive multitask language understanding in chinese. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 11260–11285, 2024.

[54] Abhinav K. Adduri, Dhruv Gautam, Beatrice Bevilacqua, Mohsen Naghipourfar, Alishba Imran, Rohan Shah, Noam Teyssier, Rishi Verma, Christopher Carpenter, Basak Eraslan, Francis Chalissery, Rajesh Ilango, Vishvak Subramanyam, Chiara Ricci-Tam, Sanjay Nagaraj, Aidan Winters, Mingze Dong, Stefanie Fellinger, Adam Krejci, Tilmann Burckstummer, Sravya Tirukkovular, Jeremy Sullivan, Brian S. Plosky, Nicholas D. Youngblut, Jure Leskovec, Luke A. Gilbert, Silvana Konermann, Patrick D. Hsu, Alexander Dobin, Dave P. Burke, Hani Goodarzi, and Yusuf H. Roohani. Predicting cellular responses to perturbation across diverse contexts with State. *Cell*, 189(19):5914–5931.e20, 2026.

[55] Yaowei Zheng, Richong Zhang, Junhao Zhang, YeYanhan YeYanhan, and Zheyan Luo. LlamaFactory: Unified Efficient Fine-Tuning of 100+ Language Models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations)*, pages 400–410, 2024.

[56] Leandro von Werra, Younes Belkada, Lewis Tunstall, Edward Beeching, Tristan Thrush, Nathan Lambert, Shengyi Huang, Kashif Rasul, and Quentin Gallouédec. TRL: Transformer Reinforcement Learning. `https://github.com/huggingface/trl`, 2020.

[57] Malte D Luecken, Maren Büttner, Kridsadakorn Chaichoompu, Anna Danese, Marta Interlandi, Michaela F Müller, Daniel C Strobl, Luke Zappia, Martin Dugas, Maria Colomé-Tatché, et al. Benchmarking atlas-level data integration in single-cell genomics. *Nature Methods*, 19(1):41–50, 2022.

[58] Alex Cloud, Minh Le, James Chua, Jan Betley, Anna Sztyber-Betley, Sören Mindermann, Jacob Hilton, Samuel Marks, and Owain Evans. Language models transmit behavioural traits through hidden signals in data. *Nature*, 652(8110):615–621, 2026.

[59] Charlotte Bunne, Yusuf Roohani, Yanay Rosen, Ankit Gupta, Xikun Zhang, Marcel Roed, Theo Alexandrov, Mohammed AlQuraishi, Patricia Brennan, Daniel B Burkhardt, et al. How to build the virtual cell with artificial intelligence: Priorities and opportunities. *Cell*, 187(25):7045–7063, 2024.

[60] Yusuf H Roohani, Tony J Hua, Po-Yuan Tung, Lexi R Bounds, Feiqiao B Yu, Alexander Dobin, Noam Teyssier, Abhinav Adduri, Alden Woodrow, Brian S Plosky, et al. Virtual Cell Challenge: Toward a Turing test for the virtual cell. *Cell*, 188(13):3370–3374, 2025.

A Appendix Texts

A.1 The Uncertainty Structure of Biomedical Text

This section reports the full diagnostic behind the claim made in the Introduction: dense low-confidence regions in biomedical text are strongly enriched for domain concepts, whereas matched general-domain controls show no corresponding entity enrichment.

Casting each corpus as an instruction–response pair. The quantity the diagnostic measures, $c_t = \pi_\theta(y_t^* \mid y_{<t}^*, x)$, is the same quantity BFT reweights, so it is only meaningful relative to a prompt. Each of the four corpora in Table 3 is therefore evaluated in (x, y^*) form. UltraChat, NuminaMath and our synthetic supervision have natural prompts. For PubMed Central full text, we construct x from the article title and section heading and use the section body as y^*; this construction is stated explicitly rather than treated as a natural instruction. We keep passages of at least 1024 tokens and truncate at 4096. The descriptive diagnostic uses stride-one windows of length 256 to identify sparse- and dense-low regimes; BFT training instead assigns each target token the centered local context defined in Section 3. The group thresholds are used only for diagnosis and never enter the training objective.

Window classification. Writing $\tilde{c}$ for the median of $\{c_t\}$ within a response, a token is *low-confidence* if $c_t < 0.5\,\tilde{c}$. For each window W_t we record the low-confidence fraction r_t and the longest contiguous low-confidence run L_t. Group A is $r_t \in [0.05, 0.20]$ with $L_t \leq 4$, and Group B is $r_t \geq 0.30$ with $L_t \geq 16$. Windows satisfying neither condition are simply not counted; the two groups are the extremes of a continuum, not a partition of all windows. Every threshold here is descriptive. The BFT objective in Equation (6) reads none of them, depending only on $c_{b,t}$ and $C_{b,t}^{\mathrm{loc}}$, and the sensitivity of the diagnostic to these cuts is reported at the end of this section.

Table 3: Matched diagnostic across real and synthetic biomedical text and two general-domain controls. Group-B prevalence measures how often dense-low windows occur. Entity A→B is the within-corpus change in the share of biomedical entities among low-confidence tokens; this contrast, rather than an absolute cross-corpus entity rate, is our primary quantity. Causal-B is the share of Group-B low-confidence tokens matched by the fixed causal-connective list.

Corpus	Domain / prompt	Group-B prev.	Entity A	Entity B	Causal-B
PubMed Central full text	biomedical, constructed x	20.4%	12.5%	60.9%	30.6%
Our synthetic supervision	biomedical, natural x	18.6%	11.8%	57.4%	31.3%
UltraChat-200k	conversational, natural x	2.9%	3.6%	4.1%	11.1%
NuminaMath-CoT	mathematical, natural x	7.5%	3.0%	3.2%	42.6%

Three readings of Table 3 matter. The two biomedical corpora exhibit comparable entity shifts although one is human-written and the other machine-generated. NuminaMath-CoT is the informative negative control: it produces more dense-low windows and more causal connectives than conversational text, yet its entity share remains flat, so dense uncertainty alone does not imply enrichment for biomedical entities. The biomedical annotation scheme does not test whether mathematical concepts are enriched in mathematical text. UltraChat also shows almost no A→B movement. Four additional controls test the main alternatives. Within the common 1024–2048-token band, Group-B entity share remains 58.7% on biomedical text versus 3.7% on general text. Stratified resampling by word-frequency decile and subword length reduces the biomedical–general gap in Group-B prevalence from $+15.2$ to $+10.7$ points but does not eliminate it, showing that rarity explains part, not all, of the effect. A temporally held-out recent-PMC subset gives 19.7% Group-B prevalence, close to the main corpus's 20.4%, and PubTator Central independently gives 57.2% Group-B entity share, close to SciSpacy's 60.9%. These controls address length, rarity, prior familiarity and tagger dependence separately.

Because stride-one windows overlap, we treat the window percentages as descriptive and do not regard windows as independent samples for significance testing. The CRAFT article-level control, cross-family location agreement, and downstream retention counterfactual provide independent forms of corroboration rather than token-level pseudo-replication.

Expert annotation. SciSpacy is itself a neural model trained on biomedical text, so its judgments and the base model's confidence could in principle be correlated through a shared training distribution rather than through the structure of the text. We therefore repeat the diagnostic on all 97 CRAFT articles [10], which provide expert concept annotations. We align scored-token offsets to the corpus-provided character offsets and form a binary mask from the union of manual concept spans, counting overlapping annotations once; CRAFT labels never enter Group assignment. Of the low-confidence tokens inside dense windows, 63.8% overlap a manually marked concept span, against 11.2% inside sparse windows and 18.5% over the corpus as a whole. This is external corroboration of concept enrichment, not direct ground truth for epistemic uncertainty or factual correctness.

Counterfactual: does the partition carry the effect? The measurements above are correlational. We therefore train discrete retention controls in which tokens selected by a fixed mask receive unit weight and all remaining valid tokens retain DFT's detached confidence weight. Selecting every token recovers SFT; the random mask is size-matched to the Group-B mask, and the Group-A and Group-B masks are computed once from the same pre-fine-tuning checkpoint. Each 70B variant is retrained from that checkpoint with the data, optimizer, and two-epoch schedule held fixed, then evaluated on VCWorld. Full BFT is shown separately because it uses continuous $\rho_{b,t}$ weights and the sample coefficient s_b rather than a binary mask.

Table 4: Retention counterfactual on VCWorld (70B). The four discrete controls differ only in which tokens receive unit rather than DFT weight; random selection is size-matched to Group B. Full BFT is the continuous, threshold-free reference.

Tokens retained at full weight	**VCWorld accuracy**
All tokens (SFT)	0.562
Random selection, size-matched to dense windows	0.571
Sparse-low windows only	0.508
Dense-low windows only	<u>0.674</u>
Full BFT (continuous)	**0.696**

Random retention lands near SFT and sparse-region retention falls below it, while retaining the dense windows alone recovers most of the gain. This ordering is inconsistent with a pure "retain more tokens" account because the random condition retains exactly as many.

Robustness of the partition. Sweeping the low-confidence multiplier from 0.3 to 0.7 and the Group-B (r_t, L_t) criterion over $(0.20, 8)$, $(0.30, 16)$ and $(0.40, 32)$ gives fifteen combinations. The within-corpus entity-share shift on PubMed Central remains between $+32$ and $+59$ points, so no particular descriptive cutoff is load-bearing. Repeating the frozen diagnostic with DeepSeek-R1-Distill-70B, Qwen2.5-14B-Instruct and Llama-3.1-8B-Instruct gives Group-B prevalence/entity share of 20.4%/60.9%, 19.1%/58.3%, and 22.7%/55.6%, respectively. Their Group-B locations overlap at a positional IoU of 0.61–0.68, against 0.12 for randomly placed windows with the same marginal coverage. This cross-family agreement is evidence that the partition is not merely an idiosyncrasy of one model's confidence, although it does not remove self-reference from the training procedure.

What this does not establish. The partition is defined by the base model's own confidence and is then used to train that same model, and no control removes that circularity entirely. The semantic validation instead comes from human-annotated CRAFT spans, the partition reproduces across independently trained models, and the retention counterfactual is scored on an external benchmark rather than on the confidence signal itself. A base model that is systematically overconfident in some subdomain would produce no dense-low regions there, and BFT would offer that subdomain no protection; we have not characterized where that failure begins.

A.2 Background: SFT and DFT Formulations

Supervised Fine-Tuning (SFT) Given a dataset $\mathcal{D} = \{(x, y^*)\}$ of instruction–response pairs, SFT minimizes the token-level cross-entropy loss

$$L_{\mathrm{SFT}}(\theta) = \mathbb{E}_{(x,y^*)\sim\mathcal{D}}\left[-\sum_{t=1}^{|y^*|}\log \pi_\theta(y_t^* \mid y_{<t}^*, x)\right], \qquad \ell_t := -\log \pi_\theta(y_t^* \mid y_{<t}^*, x), \tag{7}$$

so that the per-token gradient is $-\nabla_\theta \log \pi_\theta(y_t^* \mid y_{<t}^*, x)$.

Token-level RL-equivalent form. Because BFT and DFT are defined token-by-token, we work out the policy-gradient equivalent of SFT directly at the token level (rather than at the sequence level). The policy-gradient form of an RL objective $J(\theta) = \mathbb{E}_{x,y_t\sim\pi_\theta(\cdot|y_{<t},x)}[r_t(x, y_{\le t})]$ is $\nabla_\theta J(\theta) = \mathbb{E}\big[\nabla_\theta \log \pi_\theta(y_t \mid y_{<t}, x)\, r_t(x, y_{\le t})\big]$. Applying importance sampling to the per-token gradient of L_{SFT} gives

$$-\nabla_\theta \log \pi_\theta(y_t^* \mid y_{<t}^*, x) = \mathbb{E}_{y_t\sim\pi_\theta(\cdot|y_{<t}^*,x)}\left[\frac{\delta(y_t, y_t^*)}{\pi_\theta(y_t \mid y_{<t}^*, x)}\big(-\nabla_\theta \log \pi_\theta(y_t \mid y_{<t}^*, x)\big)\right], \tag{8}$$

where $\delta(y_t, y_t^*)$ is the per-token Kronecker delta. Treating the implicit per-token reward as $r_{\mathrm{SFT},t}(x, y_{\le t}) = \delta(y_t, y_t^*)$ and the implicit per-token weight as $w_t(y_t \mid y_{<t}^*, x) = \pi_\theta(y_t \mid y_{<t}^*, x)^{-1}$, the SFT gradient at every position t takes the RL-equivalent form

$$\begin{aligned}-\nabla_\theta \log \pi_\theta(y_t^* \mid y_{<t}^*, x) = -\,\mathbb{E}_{y_t\sim\pi_\theta(\cdot|y_{<t}^*,x)}\Big[&w_t(y_t \mid y_{<t}^*, x)\, r_{\mathrm{SFT},t}(x, y_{\le t}) \\ &\cdot \nabla_\theta \log \pi_\theta(y_t \mid y_{<t}^*, x)\Big].\end{aligned} \tag{9}$$

This representation makes the formal importance-weight contrast explicit at the same granularity as DFT and BFT: the *RL-equivalent* per-token weight is $w_t = \pi_\theta(y_t^* \mid y_{<t}^*, x)^{-1} = c_t^{-1}$, which is unbounded as the reference-token confidence approaches zero. This does not mean that the ordinary softmax cross-entropy gradient with respect to the logits is itself unbounded; that gradient remains bounded. DFT and BFT can instead be read as different choices of *detached* per-token loss weight on the same underlying token-level objective.

Dynamic Fine-Tuning (DFT) DFT [2] stabilizes SFT by multiplying each token-level loss with its detached predicted probability:

$$L_{\mathrm{DFT}}(\theta) = \mathbb{E}_{(x,y^*)\sim\mathcal{D}}\left[\sum_{t=1}^{|y^*|}\mathrm{sg}\big(c_t\big)\,\ell_t\right], \tag{10}$$

where $c_t = \pi_\theta(y_t^* \mid y_{<t}^*, x)$ and $\mathrm{sg}(\cdot)$ is the stop-gradient operator. Composed with the token-level RL-equivalent form above, the effective importance-sampling weight in DFT becomes $w_t = c_t^{-1} \cdot c_t = 1$, removing the inverse-probability blow-up of SFT; the DFT training objective itself still multiplies each token loss by $\mathrm{sg}(c_t)$. However, this absolute-confidence rule suppresses tokens irrespective of whether their low confidence reflects sparse stylistic noise or dense biomedical content. BFT addresses this by replacing c_t with the group-normalized weight $\rho_{b,t} = \mathrm{clip}(c_{b,t}/(C_{b,t}^{\mathrm{loc}} + \varepsilon), 0, 1)$ and by adding the sample-level coefficient s_b at the sequence boundary.

Sequence-level normalization for SFT, DFT, and BFT For a fair comparison, all three baselines in our experiments share the same outer aggregation as the BFT objective in Equation (6), namely a per-sample mean over valid tokens followed by a per-batch mean. Concretely, given a mini-batch $\{(x_b, y_b^*)\}_{b=1}^B$ with mask $m_{b,t} \in \{0, 1\}$, we use

$$L_{\mathrm{SFT}}(\theta) = \frac{1}{B}\sum_{b=1}^{B}\frac{\sum_t m_{b,t}\,\ell_{b,t}}{\sum_t m_{b,t} + \varepsilon}, \qquad L_{\mathrm{DFT}}(\theta) = \frac{1}{B}\sum_{b=1}^{B}\frac{\sum_t m_{b,t}\,\mathrm{sg}(c_{b,t})\,\ell_{b,t}}{\sum_t m_{b,t} + \varepsilon},$$

which differ from the BFT objective in Equation (6) only in the per-token weight (none for SFT, $\mathrm{sg}(c_{b,t})$ for DFT, $\mathrm{sg}(\rho_{b,t})$ for BFT) and in the sample-level coefficient (s_b for BFT, identically 1 for SFT and DFT). All three methods use the same *nominal* optimizer, learning-rate schedule, batch size, sequence length, and outer normalization. BFT differs through its confidence-derived token and sample weights; because the batch mean of s_b can vary, these weights can also change the effective gradient scale.

A.3 Data Construction, Splits and Leakage Control

Medical trajectories. HealthBench ships *Consensus* (3,671 prompts) and *Hard* (1,000 prompts) as separate files, but they are not disjoint: 586 prompt ids appear in both, that is 58.6% of *Hard*. Since we synthesize supervision from *Consensus* and evaluate on *Hard*, we remove the intersection by id first and generate candidates only from the remaining 3,085 prompts. GPT-5.1 receives the clinical instruction and its complete Consensus rubric and generates only the response trajectory. Sequential filtering retains 2,820 trajectories (91.4%; mean length 1,842 tokens). The id removal precedes generation, so no *Hard* evaluation prompt is shown to the synthesis model.

Biological instruction pairs. Source passages are the NCBI-derived gene and protein summaries distributed with GenePT. EasyDataset segments each passage and prompts GPT-OSS-120B for instruction–response pairs; the prompt supplies the passage, asks for questions answerable from it alone, and forbids reference to the passage as an artifact ("according to the text"). Sequential filtering retains 48,236 pairs (87.6%; mean response length 1,127 tokens).

Filtering. For both sources we apply, in order, a 64-token minimum response length, MinHash near-duplicate removal at Jaccard similarity 0.85, rejection and disclaimer-template removal, and evaluation-overlap filtering. Biological responses additionally must contain at least one entity recognized by SciSpacy `en_core_sci_lg`. Both sources are rubric- or passage-conditioned synthetic supervision rather than human gold annotation; source conditioning does not itself guarantee factual correctness.

Generation templates. The medical template has the fields `Input: {clinical instruction; complete Consensus rubric}`, `Task: produce one coherent response covering every rubric dimension`, and `Output: {response only}`. The biological template has `Source: {NCBI-derived GenePT passage}`, `Task: produce one instruction-response pair grounded only in the source`, and `Output: Instruction {...}; Response {...}`. The generators are instructed not to mention the supplied rubric or passage in their outputs.

Decontamination. Independently of the id-level removal above, every evaluation set is checked against the candidate training corpus by 13-gram overlap after lowercasing and punctuation stripping. This removes 0.00% of the medical source and 0.03% of the biological source, after which no evaluation 13-gram remains. We then form a source-stratified 95:5 train–validation split with seed 42. For the representation benchmarks the split is gene-disjoint: no gene whose profile is generated at evaluation time appears in any training instruction. We report these checks separately because id removal catches shared prompts, whereas the n-gram pass catches shared content reached through different prompts.

Mixture. After filtering, the combined dataset contains 2,820 medical trajectories and 48,236 biological instruction–response pairs, for 51,056 samples in total. The medical-to-biological mixture is approximately 1:17.1 by sample count. The stratified 95:5 split places 48,503 examples in the supervised-training partition.

Control without instruction synthesis. To test whether the method merely fits teacher-generated response style, we remove the synthesis step and train directly on the underlying human-written NCBI passages as targets, matching the synthetic condition in example and token count. Absolute VCWorld accuracy is lower in this condition, but BFT still exceeds SFT by 0.118, close to the 0.134 gain with synthetic instruction data (Table 5).

Table 5: VCWorld accuracy after supervised training directly on raw NCBI text, without instruction synthesis.

Method	VCWorld accuracy
DFT	0.472
SFT	0.531
BFT	**0.649**

A.4 Training and Evaluation Configuration

Supervised stage. SFT, DFT and BFT share one recipe and differ only in the confidence-derived loss weights described in Appendix A.2. We perform full-parameter fine-tuning for two epochs in LlamaFactory using AdamW, a learning rate of 1×10^{-5}, a cosine schedule with linear warmup over the first 3% of steps, bf16 mixed precision and DeepSpeed ZeRO-3. The global batch size is 64 and the sequence-length cutoff is 4096. For the 70B runs, 32 NVIDIA H20 GPUs use one example per device and two gradient-accumulation steps. With 48,503 training examples, one epoch contains $\lceil 48{,}503/64 \rceil = 758$ optimizer steps and the two-epoch schedule contains 1,516 steps. The window size is $g = 256$ for all reported BFT runs; Figure 3b sweeps it. The numerical constant is $\varepsilon = 10^{-8}$.

GRPO stage. GRPO is run for one epoch with the TRL `GRPOTrainer`, AdamW, bf16 and DeepSpeed ZeRO-3 on H20 GPUs. The shared default uses learning rate 5×10^{-7}, KL coefficient $\beta = 0.05$, group size $G = 8$, sampling temperature 0.7 and at most 2048 new tokens. Differential-expression status and direction are combined into a three-class target: Up, Down, or No (no differential expression). The task reward is one when the predicted class matches the Tahoe-100M-derived reference label and zero otherwise; a 0.1 format-validity reward is added when the output contains a parseable prediction. There is no length or process reward. We also apply the same 3×3 learning-rate–β grid to every initialization, with G fixed. Table 1 reports the shared default; grid search improves SFT and DFT to 0.517 and 0.447 but leaves both below their pre-GRPO values of 0.562 and 0.490, while BFT reaches 0.741 from 0.696. The reported grid therefore reduces dependence on one default, but does not rule out other optimization settings or establish a universal initialization ranking.

Interpreting post-GRPO declines. GRPO uses group-relative rewards on training prompts, not a guarantee of improvement on held-out cell lines. Finite-sample updates, the composite reward and distribution shift can yield lower test accuracy even if training reward increases. Table 1 reports held-out performance; it does not demonstrate a decrease in training reward, optimization collapse, or a specific mechanism of knowledge loss.

Decoding and variance. Unless stated otherwise, generation follows the VCWorld defaults (temperature 0.6). In Table 1 and Figures 2–3, error bars denote mean $\pm$ standard deviation over eight stochastic inference passes of one fine-tuned checkpoint on the same prompts; they quantify decoding variability only. In contrast, the count-valued entries in Table 10 are mean $\pm$ standard deviation across its 20 prompts after deterministic metric extraction. Fine-tuning variance is reported separately below and must not be inferred from either type of error bar.

Table 6: VCWorld average accuracy across three independent fine-tuning seeds. The final column is the standard deviation across the three seed-level means, not decoding variability.

Method	Seed 1	Seed 2	Seed 3	SD
DFT 70B	0.490	0.466	0.508	0.021
SFT 70B	0.562	0.541	0.578	0.019
BFT 70B	0.696	0.679	0.708	0.015

The three ranges are disjoint in these runs; the worst BFT seed (0.679) remains 0.101 above the best SFT seed (0.578). The paired tests in Section 5 use the five held-out cell-line means and therefore test consistency across cell lines rather than across only three training seeds.

Compute. Local training and evaluation of the open-weight models use NVIDIA H20 GPUs; third-party API calls for synthetic-data generation and benchmark baselines are separate. One 70B supervised post-training run takes approximately 36 hours on 32 H20 GPUs, or about 1,150 H20 GPU-hours; one 70B GRPO run uses approximately 770 H20 GPU-hours. These are per-run estimates rather than an audited total across every seed, ablation, hyperparameter setting, and evaluation. The diagnostic passes are inference-only and require one teacher-forcing pass per passage without gradient computation.

Sequence-length and mixture sensitivity. Shortening the training cutoff from 4096 to 2048 reduces BFT's VCWorld accuracy from 0.696 to 0.671. This dependence is expected because

$s_b = 1 - \min_t C^{\text{loc}}_{b,t}$ depends on how many local windows a sequence contains even when g is fixed. Changing the medical:biological ratio from the default 1:17.1 to 1:8 and 1:1 gives 0.691 and 0.677, respectively. The ordering remains useful across these settings, while the default long-context, biology-heavy mixture performs best.

A.5 Controls for Sample-Weight Scale and Sequence Length

The minimum-based coefficient $s_b = 1 - \min_t C^{\text{loc}}_{b,t}$ has two potential confounds. First, its batch mean changes the effective gradient scale. Second, a longer response exposes more candidate windows and therefore has more opportunity to attain a small minimum. We isolate these effects with two controls while leaving token weights, data, optimizer, and all other settings unchanged.

Batch-normalized sample weighting. We normalize the original coefficient to unit batch mean:

$$\widehat{s}_b := \frac{s_b}{\frac{1}{B}\sum_{j=1}^{B} s_j + \varepsilon}. \tag{11}$$

This preserves relative allocation across samples while removing batch-to-batch variation in the mean sample weight.

Fixed-count local-window weighting. For every response, we choose the same $K = 8$ approximately evenly spaced valid token positions, denoted S_b, and define

$$s_b^{(K)} := \text{sg}\left(1 - \min_{t \in S_b} C^{\text{loc}}_{b,t}\right), \qquad |S_b| = K. \tag{12}$$

Because every response contributes the same number of candidate local contexts, this control removes the direct extreme-value advantage of longer responses while retaining sensitivity to a difficult region. All filtered training responses contain at least 64 target tokens and therefore support eight distinct positions.

Table 7: Sample-level controls on VCWorld (70B). All rows use the same supervised recipe. Best in each column is shown in **bold**.

Sample-level rule	VCWorld accuracy
No sample coefficient (BFT-token)	0.660
Original minimum coefficient	<u>0.696</u>
Batch-normalized coefficient $\widehat{s}_b$	**0.700**
Fixed-count coefficient $s_b^{(K)}$, $K = 8$	0.690

Batch normalization slightly improves accuracy from 0.696 to 0.700, providing no evidence that BFT's gain depends on changing the mean gradient scale. Fixed-count sampling reaches 0.690, retaining 95.5% of the original BFT–SFT gain and remaining 0.030 above BFT-token. Thus, the sample-level benefit persists when every response contributes the same number of candidate local contexts; sequence length may affect the raw minimum, but it does not account for the observed improvement.

A.6 Additional Model Families

The main results use the DeepSeek-R1-Distill series. To check that the effect is not specific to one family or to distillation from one teacher, we repeat the supervised stage on two unrelated backbones under the same recipe and evaluate on VCWorld (Table 8).

Table 8: VCWorld average accuracy across model families under an identical recipe. The DeepSeek-R1-Distill row is the 70B result from Table 1; the other two families are smaller, so absolute values are lower and only the within-row ordering is meaningful.

Base model	DFT	SFT	BFT
Qwen2.5-14B	0.42	0.47	**0.57**
Llama-3.1-8B	0.37	0.42	**0.52**
DeepSeek-R1-Distill-70B	0.49	0.56	**0.70**

The ordering BFT $>$ SFT $>$ DFT holds in all three families, including the inversion in which DFT falls below plain SFT — an ordering consistent with, but not proof of, the proposed suppression mechanism. The diagnostic itself also transfers without any training: as reported in Appendix A.1, the Group A/Group B partition computed from these same three backbones agrees at a positional IoU of 0.61 to 0.68.

A.7 Comparison with Other Reweighting Objectives

Section 2 cites several recent token-level reweighting methods without evaluating them. We run four of them here under the recipe used for SFT, DFT and BFT: identical data, optimizer, schedule, sequence length and decoding, with only the objective changed (Table 9).

Table 9: VCWorld average accuracy for alternative reweighting objectives, 70B, shared recipe.

Objective	VCWorld accuracy
DFT + BFT sample coefficient	0.528
iw-SFT [21]	0.578
UFT [22]	0.591
Anchored SFT [23]	0.607
ProFit [24]	0.618
BFT (ours)	**0.696**

The four published supervised reweighting baselines improve on plain SFT and all fall short of BFT; against the strongest, ProFit, BFT gains 0.078 across the five cell lines ($p = 0.002$, exploratory paired test). Adding BFT's sample coefficient to DFT reaches only 0.528, showing that generic hard-sample emphasis does not recover the full method without local normalization. The published baselines reweight by importance, by anchoring to a reference policy, or by a fitness estimate, none of which distinguishes a low-confidence token in a confident neighbourhood from one inside an equally uncertain run. We exclude RAFT [25] because it requires an additional reward signal and cannot be placed under the same supervised-only recipe.

A.8 GRPO Compatibility Analysis

We evaluated the response characteristics of the SFT-aligned, DFT-aligned, and BFT-aligned 70B models on 20 biomedical causal reasoning prompts covering NSCLC molecular mechanisms (e.g., KRAS G12C oncogenesis, KEAP1-NRF2 chemoresistance, PD-L1 immune evasion). Each model generated responses under the same decoding configuration (temperature 0.6, max tokens 2048).

Automated evaluation metrics. To avoid an LLM-as-judge evaluation loop, we use reproducible lexical rules and deterministic SciSpacy entity recognition rather than generative-model judgments:

- **Biomedical entity count**: the number of unique biomedical mention spans detected by SciSpacy [7] (`en_core_sci_lg` model) per response. This model detects untyped mentions; the count is not a validated count of specific entity classes.
- **Causal connective count**: occurrences of causal and mechanistic language from a predefined list of 45 biomedical indicators, including action verbs ("activates", "inhibits", "phosphorylates", "upregulates"), consequence markers ("leads to", "results in", "confers

resistance to"), and pathway connectives ("downstream of", "via", "through the ... pathway").

- **Response length**: total character count; length alone does not measure mechanistic depth.
- **Drug/mutation mention rate**: fraction of responses that mention at least one specific drug name (matched against a curated list of 120 FDA-approved or clinical-stage oncology compounds) or genetic mutation (matched via regex pattern for standard HGVS-like notation, e.g., G12C, T790M, V600E).

Table 10: Automated response analysis for SFT, DFT, and BFT-aligned 70B models on 20 biomedical causal reasoning prompts. Count-valued columns report mean $\pm$ standard deviation across prompts, not repeated inference runs. **Entities**: unique biomedical entities per response (SciSpacy `en_core_sci_lg`). **Causal**: causal connective count from a 45-term indicator list. **Length**: response character count. **Drug/Mut.**: fraction of responses mentioning $\geq$1 specific drug or mutation. Metric extraction is deterministic conditional on the generated responses.

Method	Entities	Causal	Length	Drug/Mut.
SFT 70B	32.4 ± 8.2	8.6 ± 2.4	3187 ± 557	75%
DFT 70B	28.1 ± 9.5	7.2 ± 2.8	2756 ± 549	65%
BFT 70B	$\mathbf{55.4} \pm 7.1$	$\mathbf{15.8} \pm 3.1$	$\mathbf{5458} \pm 635$	**95%**

BFT produces responses with 71% more biomedical entities than SFT and 97% more than DFT, and its causal connective count nearly doubles (+84% over SFT, +119% over DFT).

These raw counts should not be read as greater entity *density*. Per thousand characters, unique-entity counts are 10.17, 10.20, and 10.15 for SFT, DFT, and BFT, respectively; BFT responses are longer and contain more entities in absolute terms, not per unit of text. Two other metrics are less dependent on length. Causal connectives per thousand characters are 2.70 for SFT, 2.61 for DFT and 2.89 for BFT. The drug/mutation mention rate is a per-response binary: 95% for BFT against 75% and 65%, although a longer response still has more opportunity to contain a mention, so the explicitly length-matched analyses below remain primary.

Rare entities and causal depth. We define rare entities as reference-answer entities in the lowest PubMed mention-frequency decile and measure whether they appear in the generated response. Recall is 0.41 for SFT and 0.63 for BFT, remaining 0.60 when BFT is length-matched to SFT's mean 3,187 characters. Automatically extracted dependency-path causal chains have mean depth 2.3, 2.1, and 3.6 for SFT, DFT, and BFT, respectively, and 3.3 for length-matched BFT. Together with the per-character connective rates above, these controls show that the ordering is not solely a consequence of BFT generating more characters, although they remain automated proxies rather than expert factuality judgments.

Implications for GRPO compatibility. These response statistics describe differences between initializations, not their causal effect on RL. The GRPO task reward is outcome-based: it does not directly score intermediate entities or causal links. A richer response therefore need not receive a better reward. The observed association with subsequent improvement should not be interpreted as evidence of improved intermediate credit assignment.

A.9 Length-Controlled Evaluation

BFT responses are longer than SFT and DFT responses (Table 10), which raises the question of whether its advantage is a length effect rather than a content effect. This section tests that directly.

Protocol. A hard truncation may remove the final prediction needed for VCWorld scoring. We therefore use a reasoning budget with forced completion: generation stops at token N; if no prediction has been stated, a fixed answer-eliciting instruction is appended and at most 32 additional tokens are decoded. The budget and rule are shared across methods. Thus, the maximum generated output is $N + 32$, not strictly N tokens, and this control does not impose identical realized lengths.

The 2048-token row numerically matches the main results. Similar accuracy at 4096 tokens suggests limited aggregate sensitivity to increasing the cap; it does not establish that no responses were truncated. The completion instruction is an intervention whose effect may differ across methods.

Table 11: VCWorld average accuracy (70B) under a shared reasoning budget N and up to 32 additional forced-completion tokens. The 2048-token row matches Table 1; 4096 tests sensitivity to a larger cap.

Reasoning budget N	**DFT**	**SFT**	**BFT**
256 tokens	0.441	0.512	**0.598**
512 tokens	0.462	0.537	**0.641**
1024 tokens	0.481	0.554	**0.672**
2048 tokens (main setting)	0.490	0.562	**0.696**
4096 tokens	0.491	0.564	**0.698**

At $N = 256$ with up to 32 completion tokens, BFT leads SFT by 8.6 points and DFT by 15.7 points; the ordering holds at every budget (Table 11). BFT loses more than SFT when compressed (-0.098 versus -0.050 from the 2048-token setting), so part of its unconstrained gain accompanies longer reasoning. The 2048 and 4096 rows are nearly unchanged, indicating small aggregate sensitivity to increasing the cap. We therefore conclude only that the BFT advantage survives stringent length control, not that response length is irrelevant.

Representation side. The same question applies to the embedding results, where a longer gene profile could produce a richer embedding for reasons unrelated to biological content. We cap BFT profile generation at SFT's mean output length and re-encode the newly generated profiles with the same TEXT-EMBEDDING-ADA-002. Averaged over the six gene-level columns, BFT moves from 0.829 to 0.818, still above GenePT's 0.802. This is a generation-time control rather than post-hoc profile truncation.

A.10 Targeted KB Support Check

We conduct a deliberately narrow automated support check on the same 20 NSCLC causal-reasoning trajectories used in Appendix A.8, one response per prompt per method. This analysis is intended to test whether the reported ordering reverses on readily checkable biomedical relations; it is not an exhaustive factuality evaluation and includes no human adjudication.

For the length-matched text-property control, we truncate each BFT response prefix to SFT's mean response length of 3,187 characters before extracting triples. Prefix truncation is appropriate for a metric computed on the retained text; it is not used for VCWorld accuracy, where removing the final prediction would invalidate scoring and the forced-completion protocol of Appendix A.9 is used instead.

What the metric measures. Relation triples are extracted from each response, and a triple enters the checkable subset only when *both* endpoints map to a SemMedDB concept or a DrugBank identifier. The reported rate is the fraction of that subset whose relation is supported by the knowledge base (Table 12). It is a support rate on an alignable subset, not a coverage figure over all claims, and it should not be read as one.

Table 12: Targeted knowledge-base support check for extracted relation triples from 20 NSCLC trajectories. The length-matched row prefix-truncates BFT responses to SFT's mean 3,187-character length before extraction. Rates apply only to the KB-alignable subset.

Method	**Triples**	**KB-alignable**	**Support rate**	**Adversarial reading**
SFT 70B	608	209 (34.4%)	71.3%	24.5%
DFT 70B	579	193 (33.3%)	68.9%	23.0%
BFT 70B	951	328 (34.5%)	**82.6%**	**28.5%**
BFT 70B, length-matched	641	215 (33.5%)	81.4%	27.3%

The alignable fraction is similar across methods (33.3–34.5%), but similar coverage does not ensure comparable claim difficulty or eliminate subset-selection bias. Counting every unalignable triple as unsupported preserves the ordering in the last column; this sensitivity analysis is not an estimate of factual accuracy over all claims.

What falls outside, and why. Roughly two thirds of extracted triples are not alignable, and the attrition is on the entity side rather than the relation side: a triple's endpoints are open text spans, and one unmappable endpoint removes the whole triple. Assigning each excluded triple to the first applicable reason, about 61% have at least one endpoint with no knowledge-base identifier at all (cell lines, assay and method names, compound codes, recently assigned gene symbols); about 24% have an endpoint that is a composite process or state phrase rather than a single concept, such as "loss of ERK phosphorylation"; and about 15% have an endpoint that is a quantitative or experimental object such as a fold-change or an IC50 value. A triple whose endpoints both map but whose relation has no counterpart in the knowledge base is counted as unsupported rather than excluded, so the rates above are conservative in that respect.

This is the principal limitation of the check. SemMedDB is derived through the automatic SemRep pipeline, whereas DrugBank is curated; their role here is to provide a check independent of the SciSpacy-based descriptive analysis. The present analysis does not establish the factuality of the unalignable residue or of generated biomedical responses in general.

A.11 More Gene-Level and Cell-Level Results

Results in Tables 13 and 14 show that standard SFT and DFT remain relatively weak for this style of profile-derived biological representation, often underperforming scGPT and sometimes even PCA. DFT exhibits the largest degradation, consistent with the concern that suppressing low-confidence tokens can also remove reasoning chains and rare entities that matter for functional profile texts. In contrast, BFT achieves the best average gene-level score and the best evaluated clustering metrics (ARI, AMI, ASW), but not the best score on every gene-level task. The comparison against GenePT holds the OpenAI TEXT-EMBEDDING-ADA-002 encoder fixed but contrasts GenePT's NCBI summaries with profiles generated by our BFT-aligned 70B model; source selection and profile construction also differ, so the difference cannot be attributed solely to the generating model. BFT 70B nevertheless matches or exceeds GenePT on all gene-level and cell-level metrics except Lys4. This advantage is also robust to the choice of encoder: under the smaller BioBERT encoder the absolute scores drop and BFT's relative gain over SFT shrinks (the experiment does not isolate encoder capacity, truncation, or pooling effects), but the ordering DFT $<$ SFT $<$ BFT remains unchanged on every column. Overall, these results are consistent with the view that BFT improves the specificity of generated biomedical profile texts in a way that benefits downstream representation learning.

Table 13: Quantitative evaluation of biological representations at the gene level. We compare the performance of embeddings derived from different fine-tuning strategies (SFT, DFT, and BFT) across two text-embedding encoders. We use OpenAI TEXT-EMBEDDING-ADA-002, the encoder used by GenePT for NCBI gene summaries; BFT instead generates profile text before encoding. This comparison holds the encoder fixed, not the source text. BioBERT is included as a domain-specific encoder. All values are reported as AUC. Bold indicates the best performance in each column.

Encoder	Method	TF Range	Dosage	No-methy	Lys4	GGI	PPI
-	scGPT	0.557	0.853	0.804	0.847	0.635	0.661
-	GenePT	0.585	0.891	0.910	**0.939**	0.806	0.681
ada-002	DFT	0.519	0.808	0.761	0.793	0.598	0.604
	SFT	0.534	0.838	0.769	0.812	0.616	0.622
	BFT (Ours)	**0.629**	**0.913**	**0.927**	0.910	**0.823**	**0.769**
BioBERT	DFT	0.495	0.806	0.755	0.777	0.566	0.583
	SFT	0.505	0.836	0.763	0.798	0.584	0.596
	BFT (Ours)	0.591	0.872	0.879	0.864	0.741	0.707

Table 14: Quantitative evaluation of cell-level clustering performance across phenotype and cell type labels. We compare embeddings derived from SFT, DFT, and BFT using TEXT-EMBEDDING-ADA-002 (the encoder GenePT uses for NCBI summaries; our method encodes BFT-generated profiles) and BioBERT (a domain-specific encoder), against traditional baselines (PCA, scGPT, and GenePT). Bold indicates the best performance in each column.

		Phenotype Clustering			**Cell Type Clustering**		
Encoder	**Method**	**ARI**	**AMI**	**ASW**	**ARI**	**AMI**	**ASW**
-	PCA	0.14	0.16	0.011	0.15	0.28	0.031
-	scGPT	0.11	0.13	0.011	0.26	0.32	0.039
-	GenePT	0.13	0.12	0.019	0.30	0.46	0.041
ada-002	DFT	0.09	0.11	0.003	0.10	0.21	0.016
	SFT	0.11	0.13	0.006	0.12	0.25	0.021
	BFT (Ours)	**0.17**	**0.25**	**0.022**	**0.34**	**0.58**	**0.061**
BioBERT	DFT	0.07	0.09	0.002	0.08	0.18	0.012
	SFT	0.09	0.11	0.005	0.10	0.22	0.018
	BFT (Ours)	0.13	0.18	0.015	0.27	0.41	0.043

B Appendix Figures

B.1 Workflow of Biological Embedding

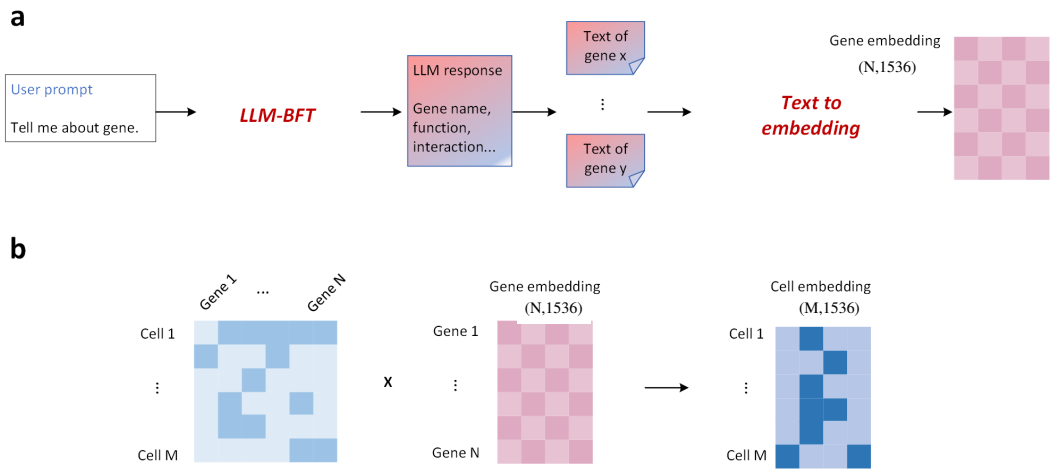

Figure 6: Workflow for extracting biological embeddings from LLM-BFT. **a**: LLM-BFT generates profile texts for entities of interest (e.g., a specific gene). The gene profile text is encoded by OpenAI TEXT-EMBEDDING-ADA-002 to obtain gene embeddings. GenePT uses this encoder on NCBI gene summaries, whereas our method encodes BFT-70B-generated profiles. **b**: For a single-cell dataset, gene embeddings are weighted by gene expression values to generate cell embeddings.

B.2 UMAP of Multi-Gene Task

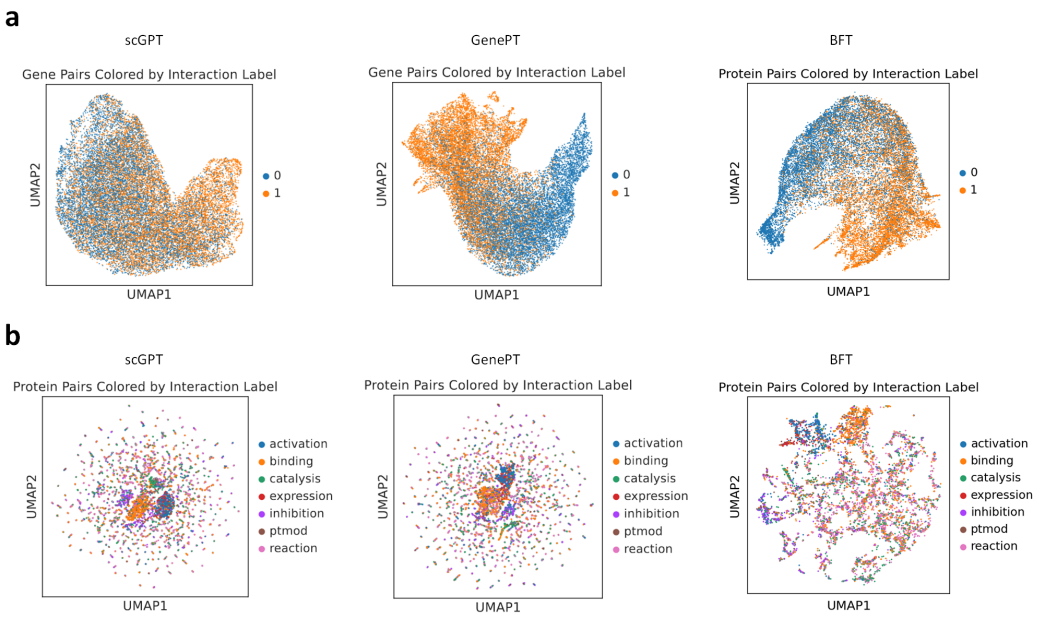

Figure 7: UMAP visualization of the multi-gene input task. **a**: For GGI, the input embedding of the classifier is directly concatenated from the embeddings of two genes. **b**: For PPI, the input embedding of the classifier is directly concatenated from the embeddings of two proteins.

B.3 UMAP of Cell Embedding

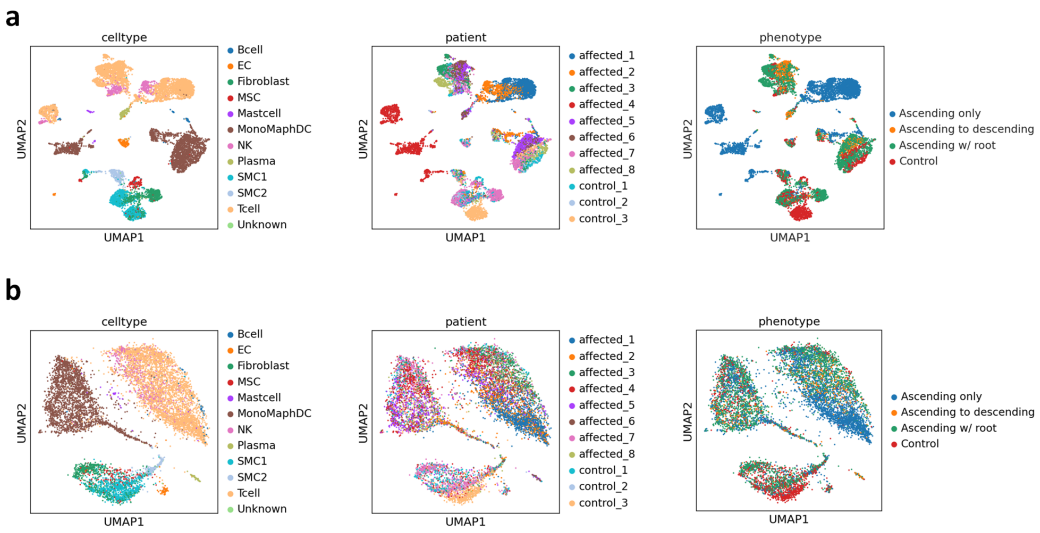

Figure 8: UMAP visualization of cell-level embeddings. **a**: PCA embeddings of the raw data, colored by cell type labels (cell type heterogeneity), patient labels (batch labels), and phenotype labels (disease heterogeneity), respectively. **b**: Cell embeddings derived from LLM-BFT, colored by cell type labels (cell type heterogeneity), patient labels (batch labels), and phenotype labels (disease heterogeneity), respectively.

B.4 UMAP of Multi-Modal Integration

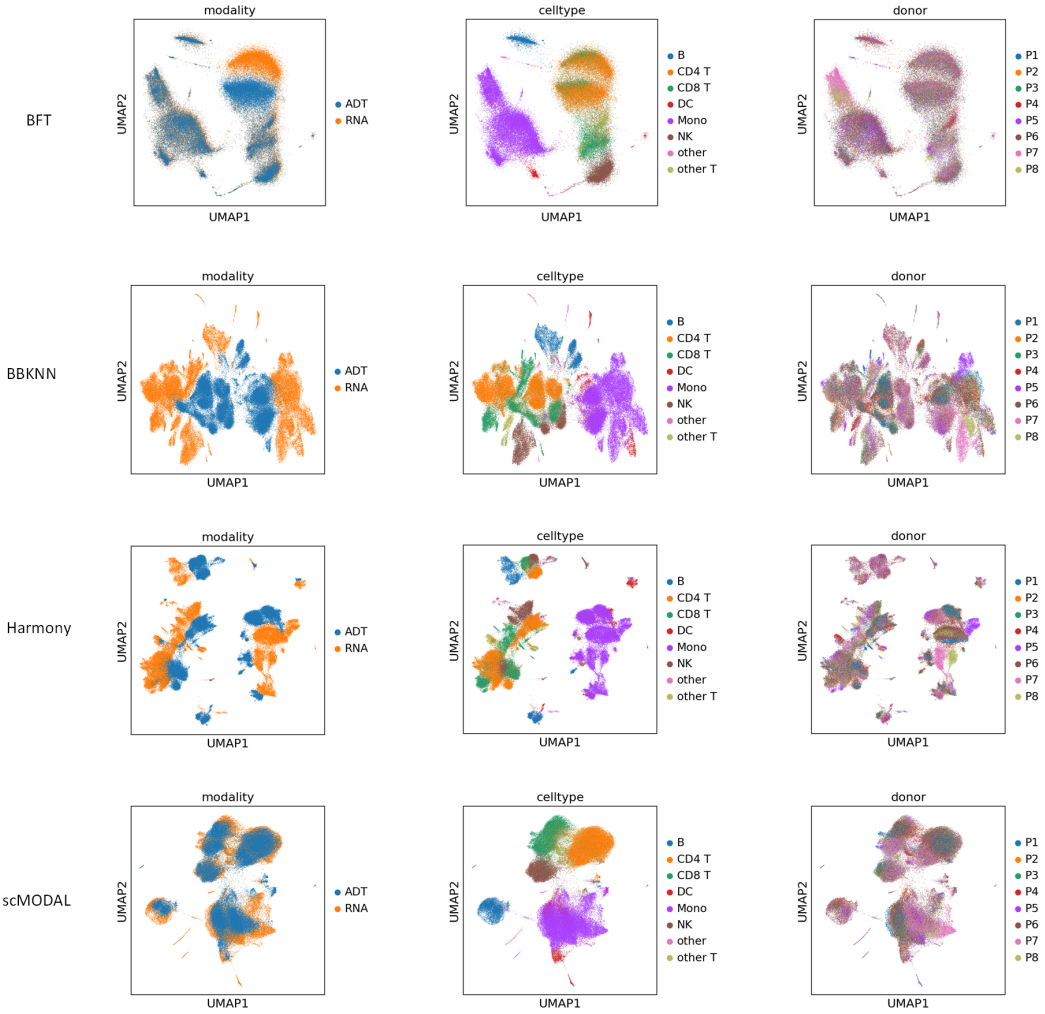

Figure 9: UMAP visualization of single-cell multimodal data integration results. Rows 1 to 4 represent different integration methods, respectively. Columns 1 to 3 correspond to different coloring labels (modality, cell type, and donor), respectively.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes].

   Justification: The abstract and introduction state the main contributions: BFT's confidence-aware objective, improvements over SFT/DFT across biomedical reasoning and sparse-reward RL, and the use of BFT-generated profile texts for downstream representations. These claims are supported by the experimental setup, results, and appendix analyses, and the conclusion states the main limitations.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes].

   Justification: The conclusion explicitly notes that BFT relies on confidence signals from the base model and synthetic biomedical supervision. It also frames the results as evidence under the evaluated biomedical settings rather than as claims of universal deployment readiness.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [N/A].

   Justification: The paper does not present formal theoretical results, theorems, or proofs. The mathematical content consists of the SFT/DFT/BFT objective definitions and optimization motivation.

   Guidelines:

   - The answer [N/A] means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes].

   Justification: The experimental setup specifies the training data sources, evaluation benchmarks, base models, baselines, backbone-replacement protocols, GRPO setting, embedding construction, and hardware family. The appendix provides the SFT/DFT formulations and additional analysis details used to interpret the main claims.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).

(d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes].

   Justification: The data sources used in the experiments are public and cited in the paper, and the implementation code is provided as supplemental material. The paper also describes the training data, model series, baselines, evaluation benchmarks, and fine-tuning framework needed to reproduce the main results.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes].

   Justification: The paper describes the datasets, task splits, base models, baselines, evaluation protocols, backbone replacements, the BFT objective including its window size parameter, and the fixed LlamaFactory training recipe. Additional appendix material gives formulation and analysis details for the main comparisons.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes].

Justification: The experimental results are reported with error bars or "±" variability values where applicable, including the main chemical perturbation reasoning table and appendix analyses. These uncertainty estimates accompany the experiments supporting the main empirical claims.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes].

   Justification: The Models and Baselines section states the base model series, fine-tuning framework, fixed optimization recipe, and compute platform used for training and inference. In particular, it specifies that all training and inference experiments were conducted on NVIDIA H20 GPUs.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes].

   Justification: The work uses public biomedical and benchmark datasets, evaluates model behavior rather than deploying a clinical system, and includes a safety-oriented analysis of hidden preference transfer. No human-subject data collection or intervention is conducted in the study.

   Guidelines:

- The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
- If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes].

    Justification: The paper discusses positive impacts for biomedical reasoning, biological profile generation, and downstream representation tasks. It also addresses safety-relevant risks through the synthetic-data hidden preference transfer experiment and notes limitations related to synthetic supervision and confidence signals.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A].

    Justification: The paper does not describe the release of a new high-risk pretrained language model, image generator, or scraped dataset. The study focuses on a fine-tuning objective and evaluations using cited public benchmarks and datasets.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.

- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes].

    Justification: The paper explicitly credits and cites the datasets, models, benchmarks, and software frameworks used, including Health Bench, GenePT, GeneAgent, VCWorld, Tahoe-100M, DeepSeek-R1-Distill, LlamaFactory, scGPT, and STATE. The study uses these existing assets according to their public availability and intended research use.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [N/A].

    Justification: The submission introduces a training objective and reports fine-tuned model evaluations, but it does not currently release a new dataset, benchmark, or model asset as part of the paper package. If models, synthetic trajectories, or code are released later, they should be documented with training details, licenses, and usage limitations.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A].

    Justification: The paper does not conduct crowdsourcing experiments or collect new data from human subjects. It relies on public datasets, model-generated synthetic trajectories, and benchmark evaluations.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A].

    Justification: The work does not involve new human-subject recruitment, intervention, or crowdsourced data collection. Therefore IRB or equivalent approval is not applicable to the reported experiments.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [Yes].

    Justification: LLMs are central to the method and experiments: the paper fine-tunes DeepSeek-R1-Distill models, uses GPT-5.1 and GPT-OSS-120B to synthesize training trajectories, and replaces GPT-4o/Gemini backbones in agent systems for evaluation. These uses are described in the introduction, experimental setup, and results.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.