HiCellS: Hierarchical Cell-Type Annotation with Self-Evolving Skill Agents

Anonymous authors
Paper under double-blind review

Abstract

Accurate cell-type annotation is fundamental to scRNA-seq analysis. Cell types are hierarchically organized from broad lineages to fine-grained subtypes, making fine-grained annotation particularly challenging. In practice, experts iteratively determine clustering granularity, interpret marker evidence, and refine cell identities hierarchically. Existing LLM-based annotators improve automation through predefined prompting workflows. However, these workflows have limited ability to revise earlier annotations based on new evidence, allowing errors in broad cell-type assignments to propagate to downstream subtype annotations. Experience from prior annotation attempts is also rarely distilled into reusable strategies across datasets. We present **HiCellS**, which casts hierarchical annotation as a learnable agent task under the guidance of a self-evolving skill bank. Initialized from teacher-generated trajectories, the agent is further optimized through reinforcement learning, using final hierarchical annotation accuracy as a reward to improve its clustering and labeling decisions. Meanwhile, successful and failed rollouts are distilled into conditional skills for adjusting clustering and interpreting marker evidence. These skills guide the corresponding decisions in subsequent rollouts and are iteratively updated from new experience, allowing the annotation policy and skill bank to co-evolve toward more accurate annotation. Across 28 scRNA-seq samples in eight dataset categories, HiCellS with a 14B backbone achieves a CL-informed matching score of 0.746, outperforming the strongest baseline by up to 0.21.

1 INTRODUCTION

Cell-type annotation is a fundamental step in single-cell RNA sequencing (scRNA-seq) data analysis, enabling the characterization of cellular composition, functional states, and disease-associated changes (Kiselev et al., 2019; Luecken & Theis, 2019). Cell types are naturally organized into hierarchical taxonomies, ranging from broad cell lineages to increasingly specific subtypes (Diehl et al., 2016). Accurate cell-type annotation therefore requires resolving cell identities consistently across multiple levels of biological granularity.

In common workflows, experts do not assign all cell identities in a single step. Instead, they iteratively adjust clustering resolution, inspect marker genes, identify broad cell lineages, and further refine selected populations into more specific subtypes (Luecken & Theis, 2019). This coarse-to-fine process adapts annotation depth to dataset-specific cellular structure, but remains labor-intensive and dependent on expert knowledge (Lähnemann et al., 2020).

Automated annotation methods reduce manual effort by assigning cell-type labels from reference expression profiles or marker genes (Abdelaal et al., 2019). Hierarchical methods further refine these assignments from broad lineages to specific subtypes (de Kanter et al., 2019; Lin et al., 2020). However, these methods generally do not place clustering, lineage assignment, and subtype refinement in a unified learnable feedback loop that can revise earlier decisions using newly obtained evidence.

Large language models (LLMs) offer a way to automate parts of this expert-driven process by using biological knowledge to interpret marker evidence and infer cell identities. Recent LLM-based annotators leverage biological knowledge and task-specific prompts to infer cell-type identities (Hou & Ji, 2024; Xie et al., 2025; Xiao et al., 2026; Song et al., 2025). These methods largely

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

rely on predefined prompts or modular workflows within each task. Some use within-task validation, repeated queries, or self-evaluation, but they have not demonstrated converting annotation outcomes and historical trajectories into continuously updated strategies or reusable skills across datasets. This limits their ability to revise ineffective annotation rules based on past outcomes, potentially allowing similar errors to persist across tasks.

To address these limitations, we present HiCellS: **Hi**erarchical **Cell**-Type Annotation with **S**elf-Evolving **S**kill Agents. HiCellS formulates annotation as an iterative agent–environment interaction. At each step, the agent observes the current annotation state, including clustering outputs, marker evidence, and biological knowledge, and selects the next annotation action. It progressively refines cell identities from broad lineages to fine-grained subtypes and can revisit previous steps when necessary. The agent is first trained with supervised fine-tuning to acquire basic annotation capabilities, followed by reinforcement learning that optimizes its actions using final annotation accuracy as feedback. Meanwhile, successful and failed rollout trajectories are distilled into conditional skills that guide the corresponding annotation steps, allowing the agent and skill bank to co-evolve (Xia et al., 2026). Our contributions are threefold:

(1) We formulate hierarchical cell-type annotation as a sequential agent learning problem, enabling adaptive coarse-to-fine refinement within a unified annotation process.

(2) We are the first to combine outcome-based reinforcement learning with step-specific skill evolution for hierarchical cell-type annotation. We improve the model's annotation decisions using final hierarchical annotation accuracy and distill the best- and worst-performing rollout trajectories into conditional skills, using ground truth as an oracle during training. These skills guide subsequent annotation decisions and are updated alongside the policy.

(3) Across 28 scRNA-seq samples in eight dataset categories, HiCellS with a 14B backbone achieves a CL-informed matching score of 0.746, outperforming the strongest baseline by up to 0.21. Ablation studies support the benefits of outcome-based policy optimization and skill evolution.

2 RELATED WORK

Reference- and Marker-Based Cell-Type Annotation. Automated annotation methods assign cell-type labels using annotated reference datasets or curated marker genes. CellTypist trains logistic-regression classifiers on reference expression profiles with harmonized cell-type labels (Domínguez Conde et al., 2022). scANVI uses a semi-supervised generative model to learn from labeled and unlabeled cells, account for batch effects, and predict labels for unannotated cells (Xu et al., 2021). scTab trains an attention-based neural classifier on large collections of cells from multiple tissues, using data augmentation to improve generalization across donors (Fischer et al., 2024). Rather than transferring labels from an expression reference, ScType assigns cell identities by combining expression scores for positive and negative marker genes (Ianevski et al., 2022). Hierarchical methods, including CHETAH and scClassify, classify cells along reference-derived cell-type hierarchies and allow predictions to stop at intermediate nodes when the evidence does not support a more specific assignment (de Kanter et al., 2019; Lin et al., 2020). Although these methods support hierarchical annotation, early clustering or lineage errors can propagate to finer subtypes. HiCellS instead uses clustering and marker evidence to adjust resolution and revisit earlier assignments during coarse-to-fine annotation.

LLM-Based Cell-Type Annotation. LLM-based methods use biological knowledge and task-specific prompts to interpret marker genes and infer cell identities. GPTCelltype provides GPT models with tissue context and cluster-level marker genes to predict cell-type labels (Hou & Ji, 2024). CASSIA combines annotation, validation, and quality-scoring agents to assess and refine marker-based predictions, with optional modules for subtype annotation and retrieval-augmented reasoning (Xie et al., 2025). CellAgent coordinates planner, executor, and evaluator agents to select analysis tools, execute analysis steps, and iteratively refine the workflow through self-evaluation (Xiao et al., 2026). Other approaches explicitly organize annotation from broad lineages to finer subtypes. scHilda first uses global knowledge-graph evidence to identify major cell types, then retrieves lineage-specific subgraphs to guide subtype annotation (Li et al., 2026). GPTAnno selects clustering resolution by evaluating ontology-based reliability and prediction consistency across repeated LLM queries, then subclusters parent populations and assigns finer labels using ontology

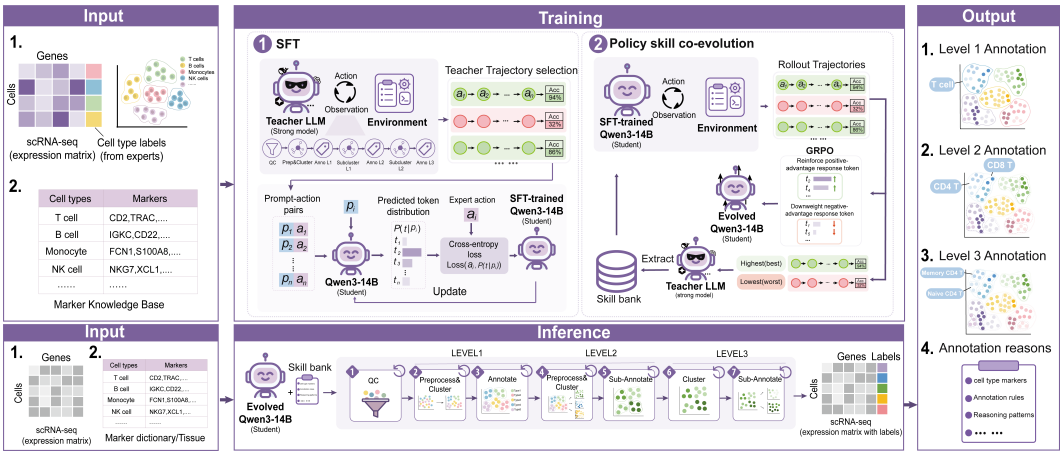

Figure 1: Skill self-evolution workflow. The base model interacts with the annotation environment, records trajectories, analyzes errors, proposes candidate skills, filters useful skills, updates the skill bank, and retrieves the evolved skills in the next annotation round.

constraints or marker inheritance (Song et al., 2025). Despite supporting iterative refinement, these methods primarily rely on predefined prompting workflows, with limited use of annotation outcomes and past trajectories to improve decisions across datasets. In contrast, HiCellS optimizes clustering and labeling decisions using final hierarchical annotation accuracy and distills rollout experience into reusable, step-specific skills that evolve alongside the policy.

3 METHODOLOGY

HiCellS turns hierarchical cell-type annotation into an agent task that mimics the workflow of human experts (Figure 1). At inference, the agent takes the same inputs used by human experts: a raw scRNA-seq expression matrix, tissue context, and a hierarchical marker dictionary. The environment presents the current annotation progress and biological evidence to the LLM as text, similar to the information inspected by an expert at each stage. Using this information and reusable annotation experience from the skill bank, the LLM decides how the analysis should proceed, as a human expert would. It selects the next action and its parameters, and the environment executes the action and returns an updated observation. Through this interaction, the agent performs adaptive quality control, clustering, and three-level annotation, producing hierarchical cell-type labels and annotation rationales.

To learn this expert-like annotation policy, training proceeds in two stages using expert cell-type labels as the ground truth. First, a strong teacher LLM interacts with the same environment, and high-accuracy trajectories are converted into input–output pairs for supervised fine-tuning (SFT). This stage teaches Qwen3-14B the interaction protocol—the input and output schemas, annotation order, skill use, and observation interpretation. Starting from the supervised-fine-tuning checkpoint, group-relative policy optimization (GRPO) improves the policy's action-selection ability using terminal hierarchical accuracy as the reward. In parallel, an experience-distillation teacher LLM evaluates the rollout trajectories turn by turn against the ground truth. It distills the experience into reusable skills and provides step-level feedback for intermediate actions. The updated skill bank guides subsequent rollouts, while the improved policy generates new experience for further skill evolution. In this way, the policy and skill bank co-evolve.

3.1 PROBLEM FORMULATION

Input and output. The input consists of a raw scRNA-seq count matrix $X \in \mathbb{N}^{n \times p}$ (n cells, p genes), tissue context, per-cell quality statistics computed upstream (doublet classification and mitochondrial-gene fraction), and a hierarchical marker dictionary $\mathcal{H}$. The dictionary specifies the admissible label space as a three-level tree: level-1 major lineages are split into level-2 subtypes,

which are further split into level-3 subtypes, and each cell type has a defining marker panel. The output is a hierarchical annotation of retained cells together with annotation rationales.

Annotation process. The process consists of quality control followed by three levels of clustering and annotation. First, the agent sets quality-control thresholds to remove doublets and low-quality cells. It then chooses a resolution and performs level-1 clustering, examines the resulting clusters and marker evidence, and assigns each cluster a broad lineage label. Within each level-1 label, the agent chooses a resolution and performs level-2 clustering, then assigns subtype labels from the corresponding child dictionary. Finally, within the level-2 labels, it chooses a resolution and performs level-3 clustering, then assigns level-3 labels.

Agent-environment interaction. At every interaction turn, the policy receives a prompt containing a fixed system instruction, the current skill bank, and an observation rendered from the pipeline state. The observation summarizes the current annotation progress, the evidence returned by the latest analysis action, the candidate labels allowed at that point, the action history, and the actions that can be executed next. The evidence includes quality summaries after filtering and, after clustering, the marker-gene expression and predefined marker-set scores for each cluster. Using both the observation and the skills, the policy selects one admissible action together with its parameters, and the environment executes the action and returns the next observation. Action selection is adaptive: each new observation informs the next action and its parameters. For example, if the level-1 clusters are poorly separated, the agent can rerun level-1 clustering at a different resolution before assigning labels. This interaction continues until the hierarchical annotation is complete.

Trajectory and formalization. A complete trajectory consists of a sequence of turns. In each turn t, the policy π_θ reads observation o_t together with skill bank $\mathcal{B}$ and selects action a_t, including its parameters. The environment executes a_t and returns the observation for the next turn. The trajectory is

$$\tau \;=\; (o_1, a_1, \ldots, o_{H_\tau}, a_{H_\tau}), \tag{1}$$

where H_τ is the number of turns. This interaction protocol is a finite-horizon Markov decision process (MDP)

$$\mathcal{M} \;=\; \langle \mathcal{S}, \mathcal{A}, \mathcal{T}, R, \mathcal{D} \rangle, \tag{2}$$

where $\mathcal{S}$ is the set of pipeline states, $\mathcal{A}$ is the set of admissible actions, $\mathcal{T}$ is the deterministic environment transition, R is the terminal reward, and $\mathcal{D}$ is the terminal condition. State s_t records the pipeline after the first $t-1$ actions, and o_t is its textual rendering. The transition $\mathcal{T}(s_{t+1} \mid s_t, a_t)$ executes a_t and updates the state. When the annotation is complete,

$$R(\tau) \;=\; \sum_{\ell=1}^{3} w_\ell \, \mathrm{Acc}_\ell(\tau), \tag{3}$$

where Acc_ℓ is the fraction of retained cells whose level-ℓ label matches the ground truth and w_ℓ is the weight assigned to that level. The objective is to learn a policy π_θ that maximizes $R(\tau)$.

3.2 Cold-Start Supervised Fine-Tuning

A pretrained LLM does not initially know how to operate the **HiCellS** environment. It must learn the input and output schemas, the annotation step order, use of skills, and interpretation of observations. We therefore use a strong teacher LLM (DeepSeek-V4-Pro) to demonstrate the interaction protocol in the same environment. Each teacher trajectory is decomposed into one input–output pair per turn, where the input is the full runtime prompt and the output is the teacher response. Given the same input, Qwen3-14B learns to imitate the teacher by maximizing the probability of generating the teacher's output tokens. We implement this objective by minimizing the negative log-likelihood,

$$\mathcal{L}_{\mathrm{SFT}} \;=\; -\,\mathbb{E}_{(x,y)\sim\mathcal{D}} \left[\sum_t \log \pi_\theta \left(y_t \mid x, y_{<t} \right) \right]. \tag{4}$$

where $\mathcal{D}$ is the set of training pairs, x is the prompt, y is the teacher response, and t indexes its tokens. Data synthesis details, statistics, and training hyperparameters are given in Appendix A.1.

3.3 Reinforcement Learning from Terminal Accuracy

Supervised fine-tuning teaches the policy to imitate teacher-demonstrated actions and operate the annotation environment, but this token-level imitation does not directly optimize the final annotation result. We therefore initialize reinforcement learning from the supervised-fine-tuning checkpoint and continue training the policy with the terminal annotation reward in Eq. 3, closing the gap between action imitation and task-level accuracy.

Group-relative policy optimization. We optimize the policy with group-relative policy optimization (GRPO) (Shao et al., 2024). For each training sample, the policy rolls out several trajectories and compares their rewards within the group. A trajectory that scores above the group average receives a positive advantage, while one that scores below the average receives a negative advantage. We compute this normalized advantage as

$$A_i \;=\; \frac{R_i - \mu_{\mathcal{G}}}{\sigma_{\mathcal{G}} + \delta}, \tag{5}$$

where $\mathcal{G}$ is the group of trajectories rolled out for one training sample, R_i is the reward of trajectory i, $\mu_{\mathcal{G}}$ and $\sigma_{\mathcal{G}}$ are the mean and standard deviation of the group's rewards, and δ is a small constant for numerical stability. GRPO increases the probability of actions in trajectories with positive advantages and decreases it for trajectories with negative advantages. To stabilize training, clipping limits overly large policy updates, while a KL penalty keeps the policy close to the supervised-fine-tuning initialization. The resulting loss is

$$\mathcal{L} \;=\; \mathbb{E}\left[-\frac{1}{G}\sum_{i}\frac{1}{|T_i|}\sum_{t}\min\left(\rho_{i,t}A_i,\; \mathrm{clip}(\rho_{i,t},\,1{-}\epsilon,\,1{+}\epsilon)\,A_i\right)\right] + \beta\,\mathrm{D}_{\mathrm{KL}}\left(\pi_\theta \,\|\, \pi_{\mathrm{SFT}}\right), \tag{6}$$

where G is the number of trajectories in a group, i indexes the trajectories, t indexes the response tokens, T_i is the set of response tokens of trajectory i, $\rho_{i,t}$ is the importance-sampling ratio between the current policy π_θ and the rollout-time policy probabilities of token t (used to correct for the off-policy nature of the rollout data), ϵ is the clipping range, and β controls the KL penalty. Training hyperparameters are listed in Appendix A.2.

3.4 Skill Evolution

Skill evolution complements outcome-based reinforcement learning in two ways. First, it distills successful and failed trajectories into reusable experience that guides action selection on future samples. Second, because the terminal reward does not directly evaluate the intermediate actions that produced it, skill evolution uses trajectory comparisons and ground-truth information to provide step-level feedback for those actions.

In our implementation, each skill has three fields: a *name*, a *principle* (a short actionable rule), and a *when-to-apply* clause (an example is given in Appendix A.3). The initial bank contains $\sim$50 hand-written rules based on expert annotation experience. During reinforcement learning, the teacher periodically distills rollout trajectories into revised skills, and the updated bank is included in the prompts of the next GRPO round.

Trajectory selection. Periodically, skill evolution runs on the trajectories rolled out in the most recent reinforcement-learning step. These trajectories are grouped by training sample, and the best and worst trajectory in each group are identified by level-3 accuracy. We selected training samples with a large performance gap between the best and the worst trajectories, because contrasting successful and failed rollouts for the same sample provides informative step-level feedback for skill distillation.

Distillation. A teacher LLM (DeepSeek-V4-Pro) analyzes every turn of the selected trajectories and distills one *skill* for the corresponding action step. For each turn, the teacher receives the prompt seen by the agent, the selected action, the resulting observation, and an *oracle* that the agent never sees. The oracle contains ground-truth statistics relevant to that action: for an annotation turn, the ground-truth label distribution of each annotated cluster; for a clustering turn, the ground-truth subtype counts within each parent. The teacher also knows whether the turn comes from the best or worst trajectory for that training sample. From a best trajectory, it summarizes the action pattern that contributed to the good outcome; from a worst trajectory, it identifies the error and the preferable

action. This turn-by-turn evaluation converts the experience into reusable skills while providing step-level feedback that is absent from the terminal reward.

Consolidation. For each annotation stage, the teacher reads the existing skills for that stage together with the newly distilled ones, then updates the skill bank: it merges duplicate skills into one, drops those that contradict the rest, and keeps the remaining unchanged. The updated bank replaces the old one and is injected into the prompts of subsequent rollouts. The teacher prompts for distillation and consolidation are given in Appendix A.4.

4 RESULTS

4.1 EXPERIMENTAL SETUP

Tasks and Datasets. We evaluate end-to-end hierarchical cell-type annotation on eight held-out scRNA-seq benchmark datasets spanning diverse tissues, disease conditions, and cellular compositions. HiCellS is trained on 113 separate samples, none of which overlap with the evaluation datasets used for supervised fine-tuning, reinforcement learning, or skill evolution. All methods use the same source datasets and reference annotations, but the evaluated cell sets differ because QC filtering is applied only to HiCellS. Baselines are evaluated without this additional QC filtering, whereas HiCellS is evaluated on its retained cells. For example, the Tabula Sapiens benchmark contains 118,067 cells after HiCellS QC, compared with 130,952 cells for the baselines. The reported comparison therefore does not use an identical cell set across methods. A single HiCellS model is evaluated across all benchmarks without dataset-specific fine-tuning. Details of the benchmark datasets are provided in Appendix B.1.

Baselines. We compare HiCellS with six representative annotation methods: CellTypist (Domínguez Conde et al., 2022), GPTCelltype (Hou & Ji, 2024), CASSIA (Xie et al., 2025), CellAgent (Xiao et al., 2026), scHilda (Li et al., 2026), and GPTAnno (Song et al., 2025), covering reference-based, LLM-based, multi-agent, and hierarchical annotation approaches. We additionally evaluate DeepSeek-V4-Pro and GPT-5.6-Sol within the HiCellS agent environment as strong API-based agent baselines. Detailed baseline methods and settings are provided in Appendix B.2.

Evaluation Metrics. We use the Cell Ontology (CL)-informed matching score throughout the overall performance comparison, ablation study, skill-evolution analysis, and parameter-sensitivity analysis to account for differences in specificity between predicted and reference cell identities. In the ablation and skill-evolution analyses, accuracy, recall, and F1 provide complementary assessments based on exact-label matching. In the parameter-sensitivity analysis, we additionally report the dataset-centered interquartile range (IQR) to summarize score variability after accounting for differences in baseline performance across datasets. Detailed scoring rules are provided in Appendix B.3.

Implementation Details. HiCellS uses Qwen3-14B as the backbone. Training and inference are implemented with verl 0.6.1 and a verl-agent fork (version 0.3.1.dev0). At inference, HiCellS uses a temperature of 0.0. The inference settings for baseline methods are specified in Appendix B.2. Data synthesis, training hyperparameters, and skill-evolution settings are detailed in Appendices A.1–A.3.

4.2 HICELLS ACHIEVES ACCURATE HIERARCHICAL ANNOTATION ACROSS DIVERSE DATASETS

As shown in Figure 2, HiCellS consistently outperforms existing annotation methods across diverse scRNA-seq datasets. HiCellS achieves the highest Cell Ontology (CL)-informed matching score on seven of eight datasets, with an average score of 0.75. The largest improvements over the best-performing baseline are observed on Tabula Sapiens, GTEx, and Vascular AS, with absolute gains of 0.29, 0.23, and 0.22, respectively. The only exception is the colon cancer dataset, on which CellTypist achieves a higher score than HiCellS. This dataset was included in the training data of the CellTypist reference model used in our evaluation, which may contribute to its stronger performance on this benchmark.

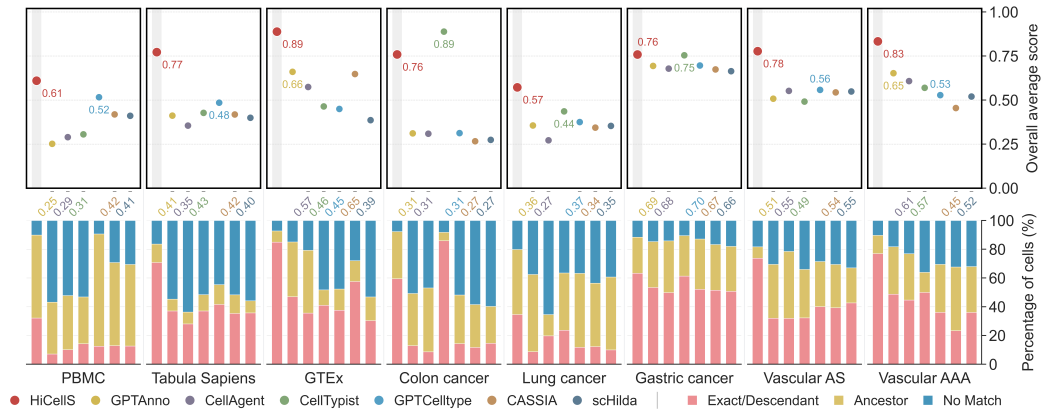

Figure 2: Cell Ontology-informed matching score across datasets. Dots summarize the average overall score for each method, and stacked bars show the fraction of exact or descendant-level, ancestral-level, and unmatched predictions.

Figure 2 further shows that HiCellS yields more exact or descendant-level matches and fewer unmatched predictions across most benchmarks, indicating closer alignment with reference cell identities under the Cell Ontology hierarchy. Summary-level results are provided in the Supplementary Material, with matching criteria described in Appendix B.3. To keep the supplementary submission within the attachment size limit, full cell-level predictions and scoring tables are not included.

4.3 Biological Consistency of Hierarchical Annotations

We further evaluated the biological consistency of HiCellS hierarchical annotations from two complementary perspectives: agreement with ground-truth (GT) subtype annotations and consistency with known marker expression. We examined these aspects using the GSE224273 atherosclerotic plaque dataset and the Kang et al. gastric cancer dataset, respectively.

In GSE224273, the predicted subtypes broadly corresponded to distinctions present in the GT annotation (Figure 3a). For example, HiCellS separated smooth muscle cells from fibromyocytes and distinguished inflammatory, foamy, and homeostatic/resident macrophages, with the predicted groups matching their corresponding GT populations. This agreement indicates that hierarchical refinement recovered reference subtype distinctions within broad vascular and immune cell classes.

In the gastric cancer dataset, HiCellS predictions were consistent with the expression of known markers for epithelial and fibroblast subtypes (Figure 3b). For example, predicted pit mucous cells expressed *MUC5AC* and *TFF1*, while matrix-remodeling fibroblasts expressed *CTHRC1* and *COL1A1*. These expression patterns provide molecular support for the identities assigned during finer annotation.

Together, these analyses show that the finer annotations reflect both reference subtype distinctions and characteristic gene-expression patterns. This complementary evidence supports the biological consistency of the coarse-to-fine annotation process and the ability of HiCellS to resolve biologically meaningful heterogeneity within broad cell populations. Additional UMAP comparisons and marker analyses are provided in Appendices C.1 and C.2, respectively.

4.4 Ablation Study

To evaluate both the contribution of each training component and the effectiveness of HiCellS relative to strong LLM agents, we compare four HiCellS variants and two API-based baselines, DeepSeek-V4-Pro and GPT-5.6-Sol, within the same agent environment. Results are averaged across the eight datasets (Table 1).

The SFT-only model achieved a CL-informed matching score of 0.711, while the RL-only variant obtained 0.258. The low score of the RL-only variant primarily stems from poor instruction follow-

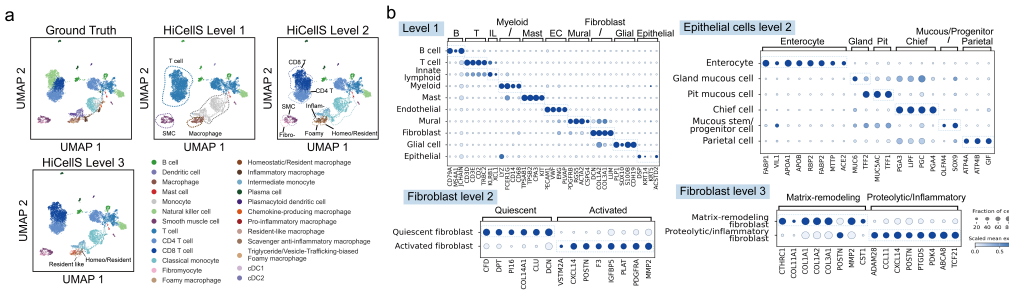

Figure 3: Biological consistency of HiCellS annotations. (a) Ground-truth and hierarchical HiCellS annotations in GSE224273. (b) Canonical-marker expression in the Kang et al. dataset. Dot size indicates expressing-cell fraction and color indicates scaled mean expression.

Table 1: API baselines and ablation results averaged over eight datasets. DS denotes DeepSeek-V4-Pro, GPT denotes GPT-5.6-Sol, and skill evol denotes skill evolution.

METHOD	AVERAGE SCORE	ACCURACY	MACRO RECALL	MACRO F1	WEIGHTED RECALL	WEIGHTED F1
HiCellS-DS (API)	0.7235	0.5377	0.4090	0.3787	0.5377	0.5366
HiCellS-GPT (API)	**0.7490**	0.5725	**0.4827**	**0.4549**	0.5725	**0.5866**
HiCellS (SFT only)	0.7106	0.5370	0.4041	0.3679	0.5370	0.5322
HiCellS (RL only)	0.2576	0.1297	0.1082	0.0960	0.1297	0.1295
HiCellS w/o skill evol	0.7335	0.5613	0.4416	0.4065	0.5613	0.5577
HiCellS	**0.7461**	**0.5766**	**0.4652**	**0.4129**	**0.5766**	**0.5598**

ing: invalid tool-call formats prevent annotation runs from completing, resulting in zero scores for most samples (see Supplementary Material for sample-level results). This highlights the role of supervised initialization in establishing the interaction capabilities needed for subsequent RL training. Initializing RL from the SFT checkpoint increased the score to 0.734, indicating that outcome-based RL improves upon the annotation capability acquired through supervised training.

Skill evolution provided an additional gain, increasing the matching score from 0.734 to 0.746 and accuracy from 0.561 to 0.577. The modest average gain partly reflects substantial declines in a few samples that offset improvements elsewhere, potentially due to more conservative subtype assignment (Section 4.5). Despite using a 14B backbone, the full HiCellS model outperformed the DeepSeek-V4-Pro API baseline and achieved a matching score comparable to GPT-5.6-Sol (0.746 vs. 0.749). These results show that supervised initialization, outcome-based RL, and skill evolution progressively improve HiCellS performance.

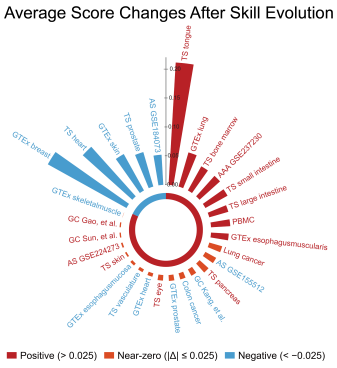

Figure 4: Sample-level changes in Cell Ontology-informed matching score after skill evolution.

4.5 Effects of Skill Evolution on Annotation

Although skill evolution yielded only a modest improvement in the overall average score, examining individual samples reveals gains in a majority of cases. (Figure 4). Specifically, eight samples improved by more than 0.025 in CL-informed matching score, 15 changed by at most 0.025 in either direction, and five declined by more than 0.025. The declines partly offset improvements elsewhere, motivating a closer examination of how evolved skills changed subtype assignment.

Improvements can arise when evolved skills help the agent recognize subtype evidence overlooked by its initial skills. In Tabula Sapiens tongue, the initial

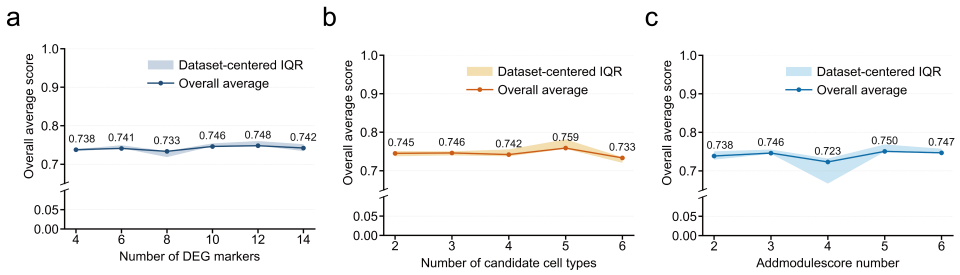

Figure 5: Sensitivity of HiCellS to annotation input settings. Average CL-informed matching scores are shown across different numbers of (a) DEG markers, (b) candidate cell types, and (c) AddModuleScore features; shading indicates the dataset-centered IQR.

skills `level2_annotation_gate_and_fallback` (Appendix A.3.1) left some basal cells labeled as epithelial cells because their subtype markers were absent from the top-ranked gene list. The evolved skills `level2_child_assignment_evidence_convergence_gate` (Appendix A.3.2) considered a broader set of marker evidence, allowing the agent to identify these cells as basal cells. Basal-cell recall increased from 0.44 to 0.87, and the sample's matching score rose from 0.5608 to 0.7725. This example illustrates how reusing annotation experience can help recover subtypes that would otherwise retain only a broad label.

Conversely, declines may occur when evolved skills make subtype assignment overly conservative. In GTEx breast, the agent more often retained the broad epithelial label instead of assigning the luminal subtype, consistent with the evolved skills requiring stronger marker support for a specific label. Luminal-cell recall fell from 0.94 to 0.31, and the matching score decreased from 0.9044 to 0.7489. This case suggests that stricter evidence requirements can leave reference subtypes unresolved and reduce annotation performance. Together, these cases illustrate the benefit of recognizing previously overlooked subtype evidence and the risk of retaining overly broad labels. Detailed skill examples and sample-level results are provided in Appendices A.3 and C.3, respectively.

4.6 Sensitivity to Annotation Input Settings

We examined how annotation performance changes with the number of DEG markers, candidate cell types, and AddModuleScore features provided to the agent. We varied the number of DEG markers from 4 to 14 and the numbers of candidate cell types and AddModuleScore features from 2 to 6, and compared the average CL-informed matching scores under these settings (Figure 5).

The average score ranged from 0.733 to 0.748 when varying DEG markers, from 0.733 to 0.759 when varying candidate cell types, and from 0.723 to 0.750 when varying AddModuleScore features. The highest scores were obtained with 12 DEG markers, 5 candidate cell types, and 5 AddModuleScore features, respectively; further increases did not improve performance. Overall, the average annotation score showed limited sensitivity to these input settings within the tested ranges.

5 CONCLUSION

To address the limited use of final annotation accuracy for improving annotation decisions and the limited reuse of annotation experience across datasets, we present HiCellS, an agent framework that combines outcome-based reinforcement learning with a self-evolving skill bank. HiCellS formulates annotation as an iterative coarse-to-fine process, allowing the agent to adjust clustering and labeling actions based on current evidence. Following supervised initialization, the policy is optimized using final hierarchical annotation accuracy as the reward. During training, step-specific ground-truth information is used to evaluate actions from the best- and worst-performing rollouts, and the resulting experience is consolidated by annotation step into reusable conditional skills that evolve alongside the policy. Experiments across 28 scRNA-seq samples spanning eight dataset categories demonstrate strong annotation performance.

AI USE STATEMENT

In this work, generative AI is an integral component of the proposed HiCellS framework. DeepSeek-V4-Pro generates annotation trajectories for supervised fine-tuning and distills and consolidates step-specific skills from reinforcement-learning rollouts, using training-time ground truth as an oracle. The annotation policy uses an LLM to select analysis actions and assign cell identities. These uses produce training trajectories, annotation predictions, and reusable skills, rather than synthetic single-cell gene-expression measurements. We also used generative AI to assist with proofreading and improving the readability of the manuscript. We have reviewed all AI-assisted work and take responsibility for the final content of this work, including text, claims, and artifacts produced with the aid of generative AI.

ETHICS STATEMENT

This work develops a computational method for hierarchical cell-type annotation using previously published single-cell RNA-sequencing datasets. We did not collect new human-subject data or access personally identifiable information, and datasets were used in accordance with their original access conditions. HiCellS is intended for biological research, not direct clinical diagnosis or medical decision making. Its annotations may reflect biases or incomplete coverage in training data, reference labels, biological knowledge sources, and the underlying language models. In particular, rare or previously unseen populations may be assigned overly broad or incorrect labels, potentially affecting downstream biological interpretation. Predicted identities should therefore be assessed together with marker expression and relevant biological context, with expert review when necessary.

REPRODUCIBILITY STATEMENT

Section 3 describes the annotation environment, supervised initialization, outcome-based reinforcement learning, and skill-evolution procedure. Appendix A.1–A.2 provides trajectory-generation procedures, training configurations, software versions, and inference settings; Appendix A.3 presents initial and evolved skill examples; and Appendix A.4 provides the teacher prompts for skill distillation and consolidation. Benchmark sources and sample composition are documented in Appendix B.1, baseline configurations in Appendix B.2, and scoring rules, label-mapping procedures, and aggregation methods in Appendix B.3. Appendix C provides additional UMAP comparisons, marker-expression analyses, and sample-level results before and after skill evolution. These details document the experimental procedures and support inspection and reproduction of the reported results.

REFERENCES

Tamim Abdelaal, Lieke Michielsen, Davy Cats, Dylan Hoogduin, Hailiang Mei, Marcel J T Reinders, and Ahmed Mahfouz. A comparison of automatic cell identification methods for single-cell RNA sequencing data. *Genome Biology*, 20:194, 2019. doi: 10.1186/s13059-019-1795-z.

Jurrian K. de Kanter, Philip Lijnzaad, Tito Candelli, Thanasis Margaritis, and Frank C. P. Holstege. CHETAH: a selective, hierarchical cell type identification method for single-cell RNA sequencing. *Nucleic Acids Research*, 47(16):e95, 2019. doi: 10.1093/nar/gkz543.

Alexander D. Diehl, Terrence F. Meehan, Yvonne M. Bradford, Matthew H. Brush, Wasila M. Dahdul, David S. Dougall, Yongqun He, David Osumi-Sutherland, Alan Ruttenberg, Sirarat Sarntivijai, et al. The cell ontology 2016: enhanced content, modularization, and ontology interoperability. *Journal of Biomedical Semantics*, 7:44, 2016. doi: 10.1186/s13326-016-0088-7.

Cecilia Domínguez Conde, Chen Xu, Laura B Jarvis, Daniel B Rainbow, Sarah B Wells, Tiago Gomes, Sarah K Howlett, Ondrej Suchanek, Krzysztof Polanski, Hannah W King, et al. Cross-tissue immune cell analysis reveals tissue-specific features in humans. *Science*, 376(6594): eabl5197, 2022. doi: 10.1126/science.abl5197.

T. Emoto, H. Yamamoto, T. Yamashita, et al. Single-cell RNA sequencing reveals a distinct immune landscape of myeloid cells in coronary culprit plaques causing acute coronary syndrome. *Circulation*, 145(18):1434–1436, 2022. doi: 10.1161/CIRCULATIONAHA.121.057923.

G. Eraslan, E. Drokhlyansky, S. Anand, et al. Single-nucleus cross-tissue molecular reference maps toward understanding disease gene function. *Science*, 376(6594):eabl4290, 2022. doi: 10.1126/science.abl4290.

F. Fasolo, G. Winski, Z. Li, Z. Wu, et al. The circular RNA ataxia telangiectasia mutated regulates oxidative stress in smooth muscle cells in expanding abdominal aortic aneurysms. *Molecular Therapy–Nucleic Acids*, 33:848–865, 2023. doi: 10.1016/j.omtn.2023.08.017. URL https://www.ncbi.nlm.nih.gov/pubmed/37680984.

D. M. Fernandez, A. H. Rahman, N. F. Fernandez, et al. Single-cell immune landscape of human atherosclerotic plaques. *Nature Medicine*, 25(10):1576–1588, 2019. doi: 10.1038/s41591-019-0590-4.

Felix Fischer, David S. Fischer, Roman Mukhin, Andrey Isaev, Evan Biederstedt, Alexandra-Chloé Villani, and Fabian J. Theis. scTab: Scaling cross-tissue single-cell annotation models. *Nature Communications*, 15:6611, 2024. doi: 10.1038/s41467-024-51059-5.

S. Gao, S. Qin, D. Wang, et al. A spatially resolved atlas of gastric cancer characterises a lymphocyte-aggregated region. *Nature Communications*, 17(1):2059, 2026. doi: 10.1038/s41467-026-68612-z.

Wenpin Hou and Zhicheng Ji. Assessing GPT-4 for cell type annotation in single-cell RNA-seq analysis. *Nature Methods*, 21(8):1462–1465, 2024. doi: 10.1038/s41592-024-02235-4.

Aleksandr Ianevski, Anil K. Giri, and Tero Aittokallio. Fully-automated and ultra-fast cell-type identification using specific marker combinations from single-cell transcriptomic data. *Nature Communications*, 13:1246, 2022. doi: 10.1038/s41467-022-28803-w.

B. Kang, J. Camps, B. Fan, et al. Parallel single-cell and bulk transcriptome analyses reveal key features of the gastric tumor microenvironment. *Genome Biology*, 23(1):265, 2022. doi: 10.1186/s13059-022-02817-9.

N. Kim, H. K. Kim, K. Lee, et al. Single-cell RNA sequencing demonstrates the molecular and cellular reprogramming of metastatic lung adenocarcinoma. *Nature Communications*, 11(1):2285, 2020. doi: 10.1038/s41467-020-16164-1.

Vladimir Yu Kiselev, Tallulah S Andrews, and Martin Hemberg. Challenges in unsupervised clustering of single-cell RNA-seq data. *Nature Reviews Genetics*, 20:273–282, 2019. doi: 10.1038/s41576-018-0088-9.

David Lähnemann, Johannes Köster, Ewa Szczurek, Davis J McCarthy, Stephanie C Hicks, Mark D Robinson, Catalina A Vallejos, Kieran R Campbell, Niko Beerenwinkel, Ahmed Mahfouz, et al. Eleven grand challenges in single-cell data science. *Genome Biology*, 21:31, 2020. doi: 10.1186/s13059-020-1926-6.

H. O. Lee, Y. Hong, H. E. Etlioglu, et al. Lineage-dependent gene expression programs influence the immune landscape of colorectal cancer. *Nature Genetics*, 52(6):594–603, 2020. doi: 10.1038/s41588-020-0636-z.

Y. Li, Y. Sun, A. Geng, et al. scHilda: Hierarchical integration of LLM with KG database for single cell type annotation. *PLOS Computational Biology*, 22(5):e1014291, 2026. doi: 10.1371/journal.pcbi.1014291.

Yingxin Lin, Yue Cao, Hani Jieun Kim, Agus Salim, Terence P. Speed, David M. Lin, Pengyi Yang, and Jean Yee Hwa Yang. scClassify: sample size estimation and multiscale classification of cells using single and multiple reference. *Molecular Systems Biology*, 16:e9389, 2020. doi: 10.15252/msb.20199389.

Malte D Luecken and Fabian J Theis. Current best practices in single-cell RNA-seq analysis: a tutorial. *Molecular Systems Biology*, 15(6):e8746, 2019. doi: 10.15252/msb.20188746.

H. Pan, C. Xue, B. J. Auerbach, et al. Single-cell genomics reveals a novel cell state during smooth muscle cell phenotypic switching and potential therapeutic targets for atherosclerosis in mouse and human. *Circulation*, 142(21):2060–2075, 2020. doi: 10.1161/CIRCULATIONAHA.120.048378.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. DeepSeekMath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.

Yiran Song, Muyao Tang, Qi Liu, Haofei Wang, Li Qian, Fei Zou, and Wenpin Hou. GPTAnno: Ontology-tree-guided hierarchical cell type annotation based on GPT models for single-cell data. *bioRxiv*, 2025. doi: 10.1101/2025.11.27.690951.

K. Sun, R. Xu, F. Ma, et al. scrna-seq of gastric tumor shows complex intercellular interaction with an alternative T cell exhaustion trajectory. *Nature Communications*, 13(1):4943, 2022. doi: 10.1038/s41467-022-32627-z.

Tabula Sapiens Consortium. The tabula sapiens: A multiple-organ, single-cell transcriptomic atlas of humans. *Science*, 376:eabl4896, 2022. doi: 10.1126/science.abl4896.

Peng Xia et al. SkillRL: Evolving agents via recursive skill-augmented reinforcement learning. In *International Conference on Learning Representations*, 2026. URL https://arxiv.org/abs/2602.08234.

Yihang Xiao, Jinyi Liu, Yan Zheng, Shaoqing Jiao, Jianye Hao, Xiaohan Xie, Mingzhi Li, Ruitao Wang, Fei Ni, Yuxiao Li, Zhen Wang, Xuequn Shang, Zhijie Bao, Changxiao Yang, and Jiajie Peng. CellAgent: LLM-driven multi-agent framework for natural language-based single-cell analysis. In *International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=BsA2GNkJhz.

E. Xie, L. Cheng, J. Shireman, et al. CASSIA: a multi-agent large language model for automated and interpretable cell annotation. *Nature Communications*, 17(1):389, 2025. doi: 10.1038/s41467-025-67084-x.

Chenling Xu, Romain Lopez, Edouard Mehlman, Jeffrey Regier, Michael I. Jordan, and Nir Yosef. Probabilistic harmonization and annotation of single-cell transcriptomics data with deep generative models. *Molecular Systems Biology*, 17(1):e9620, 2021. doi: 10.15252/msb.20209620.

Grace X. Y. Zheng, Jason M. Terry, Phillip Belgrader, et al. Massively parallel digital transcriptional profiling of single cells. *Nature Communications*, 8:14049, 2017. doi: 10.1038/ncomms14049.

A Additional Method Details

A.1 SFT Data Synthesis and Training Details

The teacher (DeepSeek-V4-Pro) runs the agent scaffold on 113 training samples, five sampled runs per sample. Per sample we keep the top-2 runs by level-3 accuracy—a rejection-sampling filter on the final outcome. This yields 226 trajectories. From each kept trajectory, only the final attempt of each turn is exported; turns that never parsed are dropped. Trajectories are split 90/10 into train/eval, giving 2,753 training and 292 evaluation turns, of which 1,658 are level-2 annotation turns, one per eligible parent. SFT uses full-parameter fine-tuning with batch size 32, maximum sequence length 16,384, learning rate 7×10^{-6}, and 3 epochs. The resulting SFT checkpoint initializes RL.

Framework and inference. The whole pipeline—SFT, GRPO rollouts, and multi-turn agent–environment interaction—is implemented and trained with verl 0.6.1 and a verl-agent fork (version 0.3.1.dev0). At inference, all models run at temperature 0.0.

A.2 Reinforcement Learning Details

RL uses terminal hierarchical annotation accuracy as the trajectory-level outcome, optimized with verl 0.6.1. Rollouts use group size G=4, temperature 0.7, and top-p 0.9. Optimization uses learning rate 5×10^{-7}, KL coefficient $\beta = 0.05$, at most 30 turns per trajectory, and 18 epochs over the 113 training samples. Skill evolution is invoked every 5 RL steps. At each invocation, training samples are ranked by the level-3 accuracy gap between their best and worst trajectories, and the top 40 samples are used for distillation. The teacher uses JSON output, a maximum of 16,384 generated tokens, and temperature 0.3.

A.3 Skill Examples

Below is a skill distilled from a level-1 annotation turn where the cluster was ambiguous between NK cell and T cell.

```
name: ambiguity-NK-vs-T
principle: use TCR markers as the primary gate to distinguish
cytotoxic T cells from NK cells: if CD3D, CD3E, or TRAC are
clearly expressed, classify the cluster as T cell even when
cytotoxic or NK-associated genes are high
when-to-apply: the candidate labels include both NK cell and T
cell, or cytotoxic markers are high
```

A.3.1 Initial Level-2 Assignment Gate

```
name: level2_annotation_gate_and_fallback
principle: Assign a configured child subtype only when parent
identity, marker expression, module score, and competing-lineage
evidence converge. If evidence is weak, conflicting, or mixed
after resolution increase, omit the cluster so it inherits the
confirmed parent label. Do not emit labels outside the active
dictionary or force rare clusters into a child subtype.
when-to-apply: before assigning any Level-2 subtype, and when
subtype evidence is insufficient
```

A.3.2 Evolved Child-Assignment Gate

```
name: level2_child_assignment_evidence_convergence_gate
principle: Assign a Level-2/3 child only within the confirmed
parent and only when the full dictionary marker panel and module
evidence converge. Ignore noisy top-ranked genes dominated by
technical or shared programs. If evidence is weak, mixed, or
supported only by isolated markers, omit the cluster so it
inherits the parent label; multiple sibling clusters may share
a child label when evidence supports it.
when-to-apply: before assigning any Level-2/3 child subtype
```

A.4 Teacher Prompts for Skill Evolution

This section gives the verbatim teacher prompts used in skill evolution (§3.4).

A.4.1 Per-Turn Distillation Prompt

```
You are a skill-summarization teacher for a single-cell RNA-seq
cell-type annotation agent.

# The agent's task
The agent controls a 7-step hierarchical annotation pipeline.
At each turn it receives a prompt (pipeline state + annotation
```

```
evidence + candidate dictionary + action list; the skill bank
is omitted here), decides ONE action (JSON), and the pipeline
executes it.

Pipeline steps:
  Step 1  QC               | decide QC thresholds (min_genes,
max_genes, max_counts, max_pct_mt)
  Step 2  Level-1 cluster   | decide leiden resolution + harmony
(only when n_samples>1)
  Step 3  Level-1 annotate  | map each cluster -> a Level-1 label
(from dictionary)
  Step 4  Level-2 cluster   | per Level-1 parent, decide
sub-clustering resolution
  Step 5  Level-2 annotate  | map each (parent, cluster) -> a
Level-2 label
  Step 6  Level-3 cluster   | per Level-2 parent, decide
sub-clustering resolution
  Step 7  Level-3 annotate  | map each (parent, cluster) -> a
Level-3 label  (terminal turn)

# Oracle
For each turn you receive an ORACLE: ground-truth (GT) information
relevant to that turn's action (e.g. the true cell-type
distribution at the level being clustered/annotated, and the GT
composition of each cluster the agent annotated). The agent never
sees the oracle; only you do. Use it to judge what the agent did
right or wrong | NOT a reward score.

# Ground-truth distribution facts (read before judging any
annotation action)
- Real tissues have SKEWED compositions: within one parent,
40-70%+ of cells being ONE child subtype is normal and correct.
Assigning the same child label to most or all sibling clusters
under a parent is CORRECT whenever the evidence and the oracle
support it.
- Rare subtypes may be genuinely tiny (tens of cells) or entirely
ABSENT from the current sample. The candidate dictionary is a
SUPERSET label space: an unused dictionary label is NOT evidence
of a missed subtype.
- Dictionary coverage and label balance across sibling clusters
are NEVER objectives. Never distill a lesson that pushes the
agent to spread labels more evenly or to "cover" unused dictionary
labels.
- Synthetic/mixed training data is artificially balanced; do not
treat balance as the natural prior.
- Before distilling any lesson like "avoid assigning the same
label to many sibling clusters" or "an unassigned dictionary label
suggests something was missed", CHECK the oracle's per-cluster GT:
if the collapsed clusters really are that true type, the action
was CORRECT | distill the opposite lesson (label frequency should
follow evidence, not balance).

# Trajectory outcome
You are told whether this turn came from the BEST or the WORST
rollout trajectory of this h5ad (by Level-3 accuracy among the 8
rollouts of the same task):
- BEST trajectory: the agent's overall line of decisions WORKED.
Distill the reusable habit that made it work (what to KEEP doing
in similar situations). Judge the turn against the oracle, but
```

```
bias toward extracting the winning procedure, not second-guessing
decisions that led to a top-scoring run.
- WORST trajectory: the agent's overall line of decisions FAILED.
Distill the mistake pattern to AVOID and, where possible, the
correct alternative the agent should have chosen instead.

# Your job
Given ONE turn (the prompt the agent saw, the action it took,
the result, the oracle, and the trajectory outcome), distill ONE
reusable skill | an experience rule that would help the agent make
a better decision in similar future situations. Reason internally
about:
- What did the agent do well or poorly THIS turn, judged against
the oracle?
- WHY did it succeed or fail | which signal in the prompt was read
correctly or misread?
- What is the generalizable lesson | what should the agent do (or
avoid) next time, and under what observable condition?

# What makes a GOOD skill (read carefully)
- GENERALIZABLE as a DECISION PROCEDURE: skills must transfer
across tissues and cell types (the agent may later face
out-of-distribution tissues). Reference signals the agent can
actually see (top_markers, module_scores, marker_expression
mean/pct, cluster pct_of_total/pct_of_parent, dictionary
eligibility, n_samples, action-history params, n_clusters).
- Dictionary-conditional EVIDENCE GATES are GOOD, even though they
name specific cell types or markers. A gate only fires when its
label is present in the dictionary, so it transfers to any tissue:
  GOOD: "Assigning the dictionary label 'CD8 T cell' requires
CD8A or CD8B support together with TCR-lineage markers
(CD3D/CD3E/TRAC); without that evidence, inherit the parent label
instead."
  GOOD: "Before assigning a rare child subtype to a large dominant
cluster, require converging child-specific evidence (module score
clear top + multiple child markers high); otherwise the dominant
cluster is almost certainly the common subtype."
- Unconditional SHORTCUTS are BAD: a marker->label mapping that
bypasses evidence weighing and would fire on any tissue.
  BAD: "If a cluster expresses CD3D, label it T cell." (no
dictionary condition, no evidence weighing)
- Concrete and actionable: name the observable condition and the
action.
- Prefer skills about: how to weigh marker expression vs module
scores vs dictionary; when to leave a cluster unmapped (inherit
parent); when to raise/lower clustering resolution; how to handle
rare/small/low-confidence clusters WITHOUT pushing label balance;
when to enable harmony; how to use action history to avoid
repeating a no-change rerun.
- Avoid skills that hard-code a specific tissue or dataset
name. Cell-type/marker names are allowed ONLY inside
dictionary-conditional gates.

# Input fields (in the user message; some may be empty)
- step                : which pipeline decision point (e.g.
step3_level1_annotation)
- is_terminal         : true if this is the final turn (Step 7
completed the trajectory)
```

```
- trajectory_outcome   : "best" | "worst" | "" (unknown) | which
rollout trajectory of this h5ad this turn came from
- agent_prompt         : the prompt the agent saw this turn (skill
bank omitted)
- agent_action         : the <action>{json}</action> the agent
emitted
- action_result        : the observation produced after executing
(next turn's prompt body; skill bank omitted). Empty on the
terminal turn.
- oracle               : GT-grounded information for this turn's
action
- final_metrics        : (terminal only) per-level accuracy /
macro-F1 of the finished trajectory

<<<USER_MESSAGE_TEMPLATE>>>
[Pipeline Step]
{{pipeline_step}}

[Is Terminal Turn]
{{is_terminal}}

[Trajectory Outcome | best or worst rollout of this h5ad by
Level-3 accuracy]
{{trajectory_outcome}}

[Agent Prompt | what the agent saw this turn (skill bank omitted)]
{{agent_prompt}}

[Agent Action | what the agent emitted]
{{agent_action}}

[Action Result | observation after executing (skill bank omitted);
empty on terminal turn]
{{action_result}}

[Oracle | ground truth for this turn's action]
{{oracle}}

[Final Metrics | terminal turn only; per-level accuracy of the
finished trajectory]
{{final_metrics}}
```

A.4.2 CONSOLIDATION PROMPT

```
You are a skill-consolidation teacher for a single-cell RNA-seq
cell-type annotation agent.

You receive, for ONE pipeline step:
1. EXISTING skills: skills already in the skill bank for this
step.
2. NEW skills: skills newly summarized from this RL step's rollout
trajectories (each distilled from one turn against a GT oracle;
each may carry an "outcome" field: "best" = distilled from the
best rollout of an h5ad, "worst" = distilled from the worst
rollout).

Your job:
1. DEDUP | merge skills that are semantically duplicate (same
rule, different wording).
```

```
2. RESOLVE CONFLICTS | if two skills contradict, keep the more
general/principled one, or merge into a conditional rule ("if X
then ..., else ...").
3. CONSOLIDATE | combine narrow/fragmented skills into fewer, more
general, more precise rules.
4. PRUNE SHORTCUTS, PRESERVE GATES |
   PRUNE: unconditional marker->label shortcuts that bypass
evidence weighing ("label CD3D+ clusters as T cell") or anything
hard-coding a tissue/dataset name. These do not transfer.
   PRESERVE: dictionary-conditional evidence gates ("assigning
label X requires marker evidence E; otherwise inherit parent").
Gates may name cell types/genes because they only fire when X
is in the dictionary | they are safety valves against systematic
misassignment, not biology hard-coding. You may rewrite a gate's
wording for clarity, but NEVER drop, weaken, or merge a gate away
into a vaguer general rule. The consolidated list must retain
every existing gate skill (in merged or equivalent form).
5. PRESERVE DISTRIBUTION NEUTRALITY | drop or rewrite any
skill that pushes dictionary coverage or label balance across
sibling clusters (e.g. "an unused dictionary label suggests
a missed subtype", "avoid assigning the same label to many
sibling clusters"). Real tissues are skewed: one child subtype
legitimately taking most cells/clusters under a parent is normal,
and the dictionary is a superset (labels may be absent). Label
frequency must follow evidence, never balance.
6. PRESERVE | keep existing skills still valid and not superseded.

Output the FINAL consolidated skill list for this step. Aim for
a concise, high-quality set (merge aggressively; prefer fewer
general skills over many narrow ones | but never at the cost of
dropping gates or admitting balance-pushing rules).

<<<USER_MESSAGE_TEMPLATE>>>
[Step]
{{step}}

[Existing Skills | already in skill bank for this step]
{{existing_skills}}

[New Skills | summarized from this RL step's trajectories]
{{new_skills}}

Output STRICTLY this JSON (no markdown, no text outside JSON):
{
  "step": "{{step}}",
  "skills": [
    {"name": "...", "principle": "...", "when_to_apply": "..."}
  ]
}

Rules:
- Output the COMPLETE final skill list for this step (merged
existing + new, deduped, conflicts resolved, shortcuts pruned,
gates preserved).
- Prefer 5-15 skills per step.
- Each principle must be concrete, actionable, and generalizable
(dictionary-conditional gates are allowed and must be kept;
unconditional shortcuts and balance-pushing rules are not).
```

B Benchmark and Experimental Details

B.1 Benchmark Datasets

Benchmark Dataset Collection and Composition. We constructed a benchmark comprising 28 h5ad samples from eight publicly available single-cell transcriptomic data collections. The benchmark includes immune-cell, normal-tissue, cancer, and vascular-disease datasets. The data sources and sample composition are summarized in Table B.1.

Table B.1: Benchmark dataset collection and composition.

Datasets	Number of samples	Included tissues or sample types
PBMC 68K	1	Peripheral blood mononuclear cells
Tabula Sapiens	10	Bone marrow, eye, heart, large intestine, pancreas, prostate, skin, small intestine, tongue, and vasculature
GTEx-derived tissues	8	Breast, esophagus mucosa, esophagus muscularis, heart, lung, prostate, skeletal muscle, and skin
Colon cancer	1	Colon cancer
Lung adenocarcinoma	1	Lung adenocarcinoma
Gastric cancer	3	Gastric tumor and related gastric tissue samples
Atherosclerotic plaque	3	Atherosclerotic plaque tissues
Abdominal aortic aneurysm	1	Abdominal aortic aneurysm
Total	**28**	—

PBMC 68K. The PBMC 68K benchmark dataset was obtained from the 10x Genomics Fresh 68K PBMC dataset and originated from the large-scale single-cell transcriptomic study by Zheng et al. (Zheng et al., 2017). This dataset contains peripheral blood mononuclear cells and is widely used as a reference dataset for evaluating cell-type annotation methods.

Tabula Sapiens. The Tabula Sapiens benchmark datasets were derived from the multi-organ human single-cell transcriptomic atlas generated by the Tabula Sapiens Consortium (Tabula Sapiens Consortium, 2022). Ten benchmark tissue-level datasets were included: bone marrow, eye, heart, large intestine, pancreas, prostate, skin, small intestine, tongue, and vasculature.

GTEx-derived tissues. The GTEx-derived benchmark datasets were obtained from the single-nucleus cross-tissue molecular reference maps generated by Eraslan et al. (Eraslan et al., 2022). Eight benchmark tissue-level datasets were included: breast, esophagus mucosa, esophagus muscularis, heart, lung, prostate, skeletal muscle, and skin.

Colon cancer. The colon cancer benchmark dataset was obtained from GEO accession GSE132465 (Lee et al., 2020). This dataset contains single-cell RNA-sequencing profiles used to characterize lineage-dependent gene-expression programs and the immune landscape of colorectal cancer.

Lung adenocarcinoma. The lung adenocarcinoma benchmark dataset was obtained from GEO accession GSE131907 (Kim et al., 2020). This dataset contains single-cell RNA-sequencing profiles used to investigate the molecular and cellular reprogramming of metastatic lung adenocarcinoma.

Gastric cancer. Three gastric cancer datasets were included in the benchmark. The **Kang et al. dataset** contains tumor and paired normal gastric tissue samples (Kang et al., 2022). The **Sun et al. dataset** was reported by Sun et al. (Sun et al., 2022), and the **Gao et al. dataset** was reported by Gao et al. (Gao et al., 2026).

Atherosclerotic plaque. Three atherosclerotic plaque datasets were included: **GSE155512**, which characterizes smooth muscle cell phenotypic switching in mouse and human atherosclerosis (Pan et al., 2020); **GSE184073**, which profiles the distinct myeloid-cell immune landscape of coronary culprit plaques causing acute coronary syndrome (Emoto et al., 2022); and **GSE224273**, which describes the single-cell immune landscape of human atherosclerotic plaques (Fernandez et al., 2019).

These datasets were grouped as atherosclerotic plaque data and counted as three independent benchmark samples.

Abdominal aortic aneurysm. The abdominal aortic aneurysm benchmark dataset was obtained from GEO accession GSE237230 (Fasolo et al., 2023). This dataset was used to investigate how the circular RNA Ataxia Telangiectasia Mutated regulates oxidative stress in smooth muscle cells in expanding abdominal aortic aneurysms.

B.2 Baseline Methods and Settings

Baseline Methods and Experimental Settings. We compared HiCellS with representative reference-based, LLM-based, and agent-based cell-type annotation methods. Each baseline was evaluated using its official implementation or workflow. CellTypist used official pretrained models with majority voting; GPTCelltype and GPTAnno performed LLM-based annotation using cluster-level marker genes; and CellAgent, CASSIA, and scHilda followed their respective agent-based annotation workflows. For methods requiring marker genes, marker lists were generated using dataset-specific clustering and differential-expression analysis. All methods followed their original annotation protocols, and their predictions were evaluated using the same Cell Ontology-informed matching procedure to ensure a consistent comparison.

Reference-based method.

CellTypist (Domínguez Conde et al., 2022). We used the official CellTypist package with its pretrained tissue-specific models and applied the recommended automatic over-clustering and majority-voting procedure to log-normalized expression matrices. The best match prediction mode was used, with the model selected according to the tissue or the closest available official pretrained model.

LLM-based methods.

GPTCelltype (Hou & Ji, 2024). We used the official GPTCelltype R implementation with the top 10 differentially expressed genes for each cluster as input. Inference was performed through an OpenAI-compatible DeepSeek endpoint with temperature 1 and top-p 1.

GPTAnno (Song et al., 2025). We used the official GPTAnno workflow for hierarchical cell-type annotation with the top 10 differentially expressed genes for each cluster as input. Inference was performed using the deepseek-chat model with 10 repeated inference runs; benchmark-provided clusters were retained in fixed-cluster experiments, while the official end-to-end workflow was used where applicable.

Agent-based methods.

CellAgent (Xiao et al., 2026). We used the official CellAgent multi-agent workflow, including its planner, executor, and evaluator components, with the top 10 differentially expressed genes for each cluster as input. Inference was performed using the DeepSeek-v4-pro model through an OpenAI-compatible API with temperature 0.

scHilda (Li et al., 2026). We used the scHilda hierarchical annotation workflow, which combines LLM-based reasoning with a curated knowledge graph and Cell Ontology information. The top 10 differentially expressed genes for each cluster were provided to the deepseek-chat model, which first generated the top three candidate cell types and then performed knowledge-graph-guided hierarchical arbitration.

CASSIA (Xie et al., 2025). We used the official CASSIA R implementation for multi-agent cell-type annotation from marker-gene tables and tissue context. The input consisted of the top 10 differentially expressed genes for each cluster; inference used deterministic generation with temperature 0 and DeepSeek-v4-flash or DeepSeek-v4-pro as specified by the corresponding benchmark run.

B.3 Evaluation Metrics

We evaluated annotation performance using a **Cell Ontology (CL)-informed matching score** as the primary metric, together with exact-label accuracy, macro-F1, and macro-recall. All methods were evaluated using the same Cell Ontology release (`format-version: 1.2`; `data-version:`

releases/2026-06-08), label-normalization procedure, ground-truth mapping, and manually curated correspondence tables.

For each cell, the ontology-informed score was defined as:

$$s_i = \begin{cases} 1, & \text{exact match or the prediction is a descendant of the ground truth,} \\ 0.5, & \text{the prediction is an ancestor of the ground truth,} \\ 0, & \text{otherwise.} \end{cases} \tag{7}$$

Thus, a correct or more specific prediction received full credit, whereas a biologically compatible but broader prediction received partial credit. Sibling labels were considered unmatched and assigned 0. The sample-level score was the mean across all cells:

$$S = \frac{N_1 + 0.5N_{0.5}}{N}. \tag{8}$$

Labels that could not be mapped automatically were resolved using dataset-specific hierarchical mappings or manually curated correspondence tables. Mixed labels, such as "T/NK cells," were mapped to multiple CL terms and scored according to their compatible components; unresolved mixed-label pairs were manually reviewed. The same frozen mapping and scoring rules were applied to all methods.

For the Cell Ontology-informed matching score reported in the main results, the cell-level counts N_1, $N_{0.5}$, and N were pooled across all samples within each dataset category before calculating the category-level score. The final score was then obtained by equally averaging the eight category-level scores. Consequently, this overall score is not the arithmetic mean of the 28 sample-level scores.

Accuracy, macro-F1, and macro-recall were calculated using strict equality between predicted and ground-truth labels, without ontology-based partial credit. Cells labeled as unassigned in the ground truth were excluded. These strict-label metrics were first computed for each of the 28 samples, averaged within each of the eight dataset categories, and then equally averaged across the eight categories.

For the parameter-sensitivity analysis, we additionally used a dataset-centered interquartile range (IQR) to quantify variability across datasets while removing differences in their absolute score levels. For dataset d and parameter setting k, the centered score was

$$\Delta s_{d,k} = s_{d,k} - \bar{s}_d, \qquad \bar{s}_d = \frac{1}{K}\sum_{k=1}^{K} s_{d,k}, \tag{9}$$

where $\bar{s}_d$ is the mean score of dataset d across all K settings of the corresponding parameter. At each setting k, we calculated the 25th and 75th percentiles of $\Delta s_{d,k}$ across the eight dataset categories. The shaded interval in Figure 5 was obtained by adding these centered quartiles to the overall average score at that setting. Thus, the interval reflects dataset-level sensitivity to parameter changes rather than differences in baseline difficulty among datasets.

C ADDITIONAL RESULTS

C.1 ADDITIONAL UMAP RESULTS

We further compared cell-type annotations on four vascular scRNA-seq datasets: three atherosclerosis datasets (GSE155512, GSE184073, and GSE224273) and one abdominal aortic aneurysm dataset (GSE237230). The UMAPs show the ground truth, six baseline methods, and the three hierarchical annotation levels produced by HiCellS using the same cell embedding.

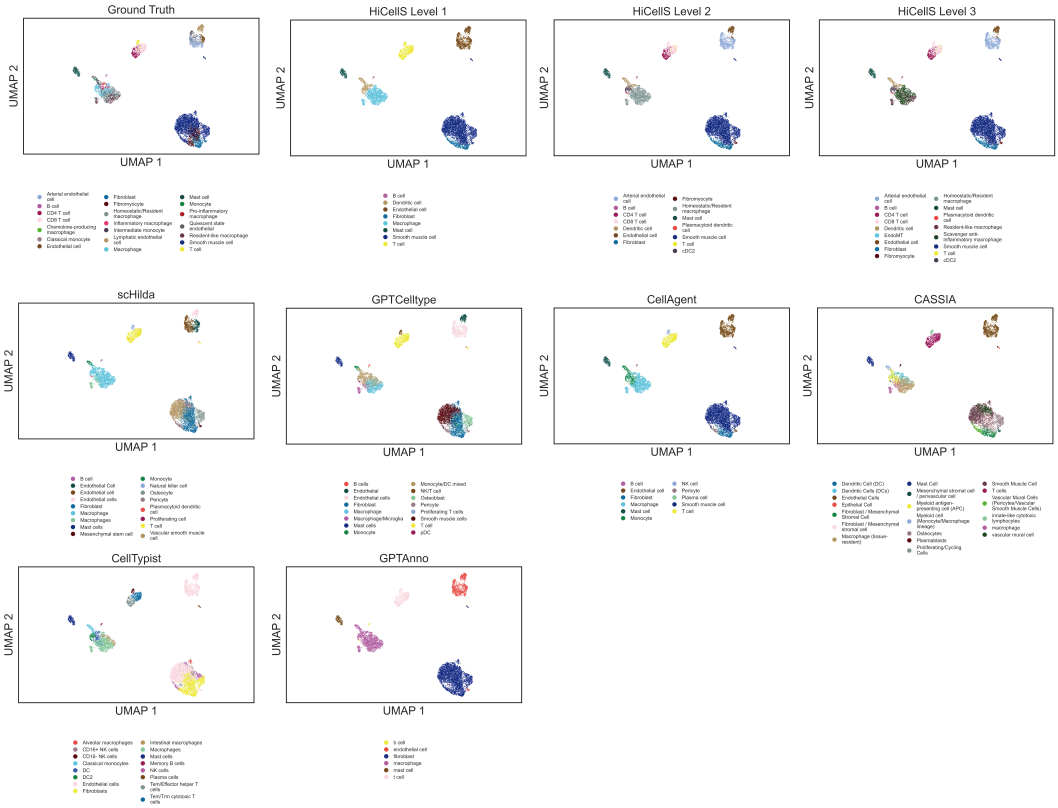

Figure C.1: UMAP comparison of ground-truth and predicted cell-type annotations on the GSE155512 atherosclerosis dataset.



Figure C.2: UMAP comparison of ground-truth and predicted cell-type annotations on the GSE184073 atherosclerosis dataset.

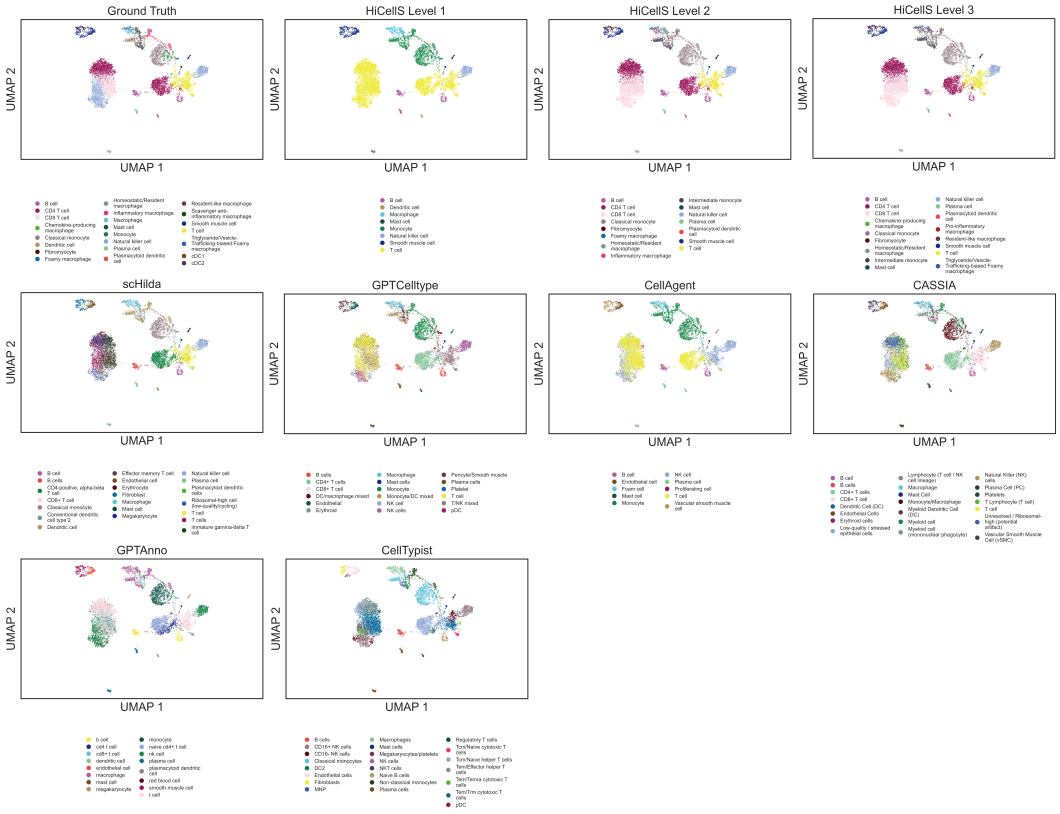

Figure C.3: UMAP comparison of ground-truth and predicted cell-type annotations on the GSE224273 atherosclerosis dataset.

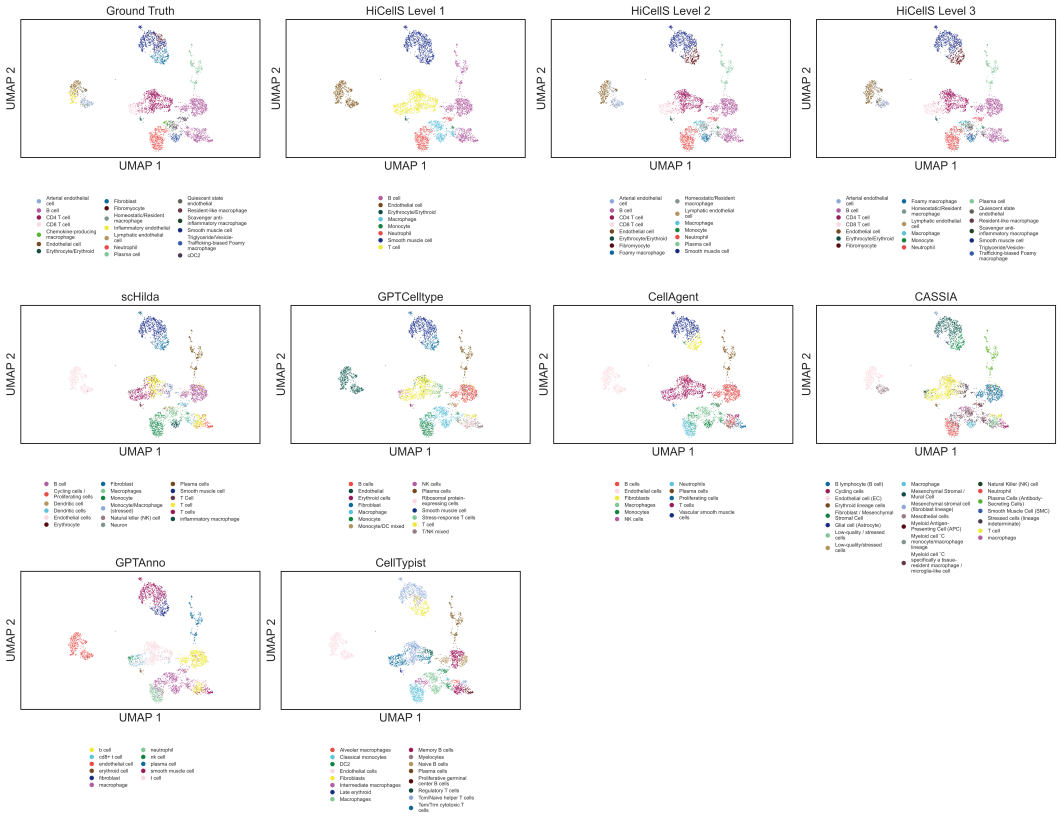

Figure C.4: UMAP comparison of ground-truth and predicted cell-type annotations on the GSE237230 abdominal aortic aneurysm dataset.

C.2 Additional Marker-expression Results

We further examined marker-gene expression patterns in the Sun et al. and Gao et al. gastric cancer datasets. These dot plots provide an additional assessment of whether the annotated cell types are supported by canonical marker expression.

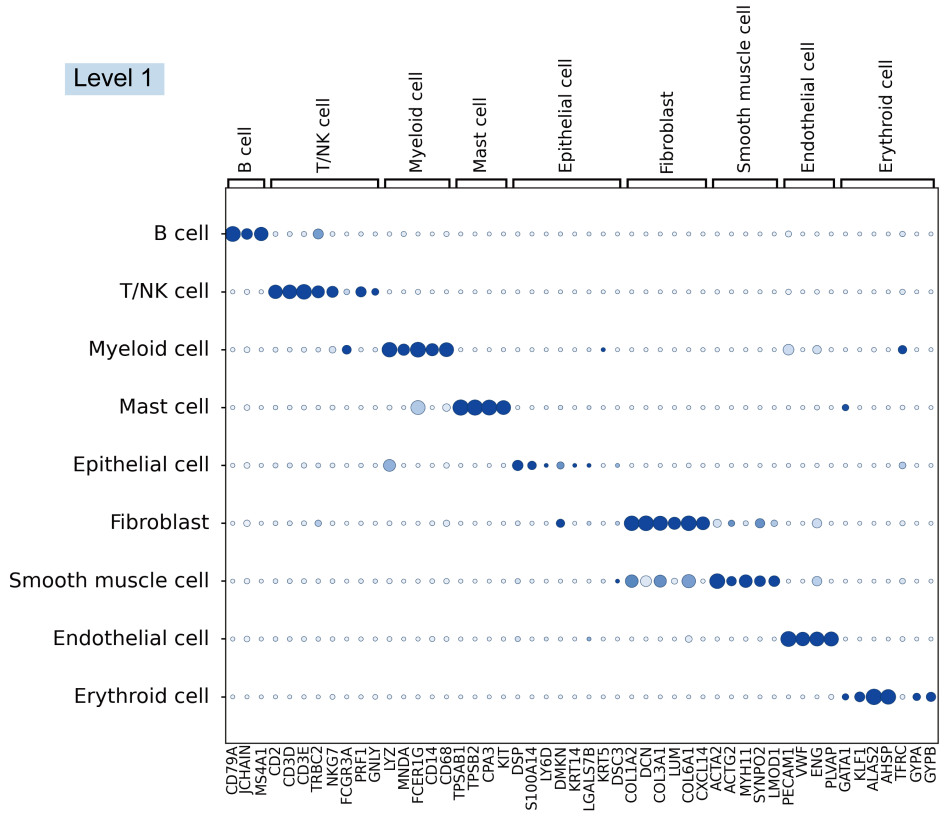

Figure C.5: Marker-gene expression patterns in the Sun et al. gastric cancer dataset. Dot size indicates the proportion of expressing cells, and color intensity represents the average scaled expression across cell types.

C.3 Detailed Skill Evolution Results

Tables C.2 and C.3 provide complementary sample-level comparisons before and after skill evolution. Table C.2 reports the sample-level Cell Ontology-informed matching scores from the non-averaged score outputs, whereas Table C.3 reports strict-label accuracy and macro recall. The main-text CL-scores of 0.733548 before skill evolution and 0.746068 after skill evolution were calculated by pooling cell-level score counts within each dataset category and then equally averaging the eight category-level scores, rather than by directly averaging the 28 rows in Table C.2. In both tables, each Δ is calculated as the value after skill evolution minus the corresponding value before skill evolution.

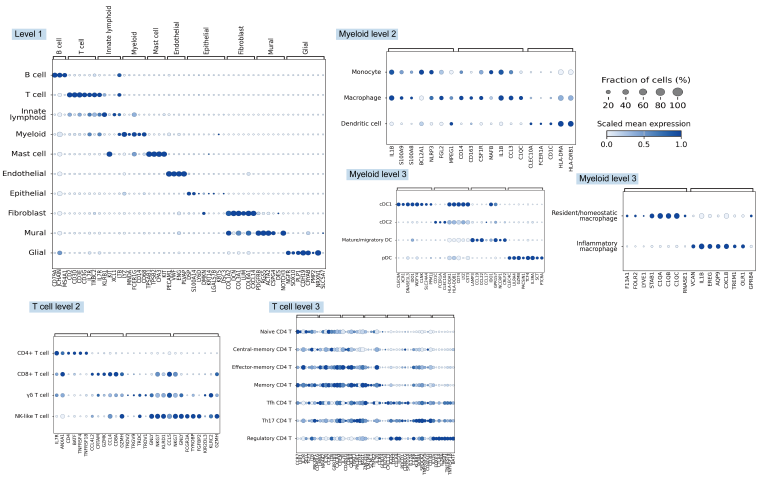

Figure C.6: Marker-gene expression patterns in the Gao et al. gastric cancer dataset. Dot size indicates the proportion of expressing cells, while color intensity represents the average scaled expression across cell types.

Table C.2: Detailed sample-level and overall Cell Ontology-informed matching scores before and after skill evolution.

Dataset	Sample	Before skill-evo score	After skill-evo score	Δ-score
PBMC 68K	PBMC 68K	0.5763	0.6096	**+0.0333**
Gastric cancer	Kang et al.	0.5752	0.5626	**-0.0126**
Gastric cancer	Sun et al.	0.9598	0.9621	**+0.0022**
Gastric cancer	Gao et al.	0.5669	0.5691	**+0.0021**
Atherosclerotic plaque	GSE155512	0.8247	0.8034	**-0.0213**
Atherosclerotic plaque	GSE184073	0.7330	0.6809	-0.0521
Atherosclerotic plaque	GSE224273	0.7735	0.7763	**+0.0028**
Abdominal aortic aneurysm	GSE237230	0.7854	0.8329	**+0.0475**
GTEx-derived tissues	breast	0.9044	0.7489	-0.1554
GTEx-derived tissues	esophagus mucosa	0.8947	0.8901	**-0.0046**
GTEx-derived tissues	esophagus muscularis	0.8978	0.9285	**+0.0306**
GTEx-derived tissues	heart	0.8365	0.8303	**-0.0062**
GTEx-derived tissues	lung	0.8914	0.9529	**+0.0615**
GTEx-derived tissues	prostate	0.8337	0.8235	**-0.0102**
GTEx-derived tissues	skeletal muscle	0.9648	0.9645	**-0.0003**
GTEx-derived tissues	skin	0.8499	0.7779	-0.0720
Tabula Sapiens	bone marrow	0.4384	0.4862	**+0.0477**
Tabula Sapiens	eye	0.8340	0.8436	**+0.0097**
Tabula Sapiens	heart	0.7100	0.5949	-0.1151
Tabula Sapiens	large intestine	0.6247	0.6589	**+0.0342**
Tabula Sapiens	pancreas	0.8616	0.8825	**+0.0209**
Tabula Sapiens	prostate	0.8788	0.8167	-0.0621
Tabula Sapiens	skin	0.8915	0.8943	**+0.0029**
Tabula Sapiens	small intestine	0.8062	0.8500	**+0.0437**
Tabula Sapiens	tongue	0.5608	0.7725	**+0.2118**
Tabula Sapiens	vasculature	0.8887	0.8833	**-0.0054**
Colon cancer	GSE132465	0.7697	0.7587	**-0.0111**
Lung adenocarcinoma	GSE131907	0.5486	0.5718	**+0.0232**
Eight-category equal-weighted score	–	0.7335	0.7461	**+0.0125**

Table C.3: Detailed accuracy and macro-recall results before and after skill evolution across the 28 benchmark samples.

Dataset	Sample	Before skill-evo accuracy	After skill-evo accuracy	Δ-accuracy	Before skill-evo macro recall	After skill-evo macro recall	Δ-recall
PBMC 68K	PBMC 68K	0.2548	0.3211	+0.0663	0.2722	0.4187	+0.1465
Gastric cancer	Kang et al.	0.3441	0.3153	-0.0288	0.2972	0.2491	-0.0480
Gastric cancer	Sun et al.	0.9598	0.9621	+0.0022	0.7810	0.7658	-0.0152
Gastric cancer	Gao et al.	0.3306	0.3154	-0.0152	0.2202	0.2854	+0.0652
Atherosclerotic plaque	GSE155512	0.6016	0.6014	-0.0003	0.3734	0.3489	-0.0244
Atherosclerotic plaque	GSE184073	0.6339	0.5970	-0.0369	0.3171	0.3285	+0.0114
Atherosclerotic plaque	GSE224273	0.6302	0.7339	+0.1037	0.4564	0.5039	+0.0474
Abdominal aortic aneurysm	GSE237230	0.6976	0.7427	+0.0451	0.5279	0.5830	+0.0552
GTEx-derived tissues	breast	0.8874	0.5722	-0.3153	0.7529	0.6702	-0.0827
GTEx-derived tissues	esophagus mucosa	0.8499	0.8423	-0.0076	0.6439	0.5880	-0.0559
GTEx-derived tissues	esophagus muscularis	0.8933	0.9284	+0.0351	0.6226	0.6947	+0.0721
GTEx-derived tissues	heart	0.6074	0.6005	-0.0069	0.6139	0.6117	-0.0022
GTEx-derived tissues	lung	0.7944	0.9174	+0.1230	0.4748	0.5262	+0.0514
GTEx-derived tissues	prostate	0.7872	0.7773	-0.0099	0.5833	0.5758	-0.0075
GTEx-derived tissues	skeletal muscle	0.9369	0.9363	-0.0006	0.5166	0.5080	-0.0086
GTEx-derived tissues	skin	0.8156	0.6823	-0.1333	0.7086	0.5467	-0.1619
Tabula Sapiens	bone marrow	0.3771	0.4327	+0.0557	0.4120	0.4836	+0.0716
Tabula Sapiens	eye	0.6786	0.7719	+0.0932	0.2531	0.3573	+0.1042
Tabula Sapiens	heart	0.6917	0.2476	-0.4441	0.5764	0.4764	-0.1001
Tabula Sapiens	large intestine	0.5544	0.6159	+0.0615	0.4196	0.3833	-0.0364
Tabula Sapiens	pancreas	0.7093	0.7892	+0.0799	0.5419	0.6049	+0.0629
Tabula Sapiens	prostate	0.6920	0.6091	-0.0829	0.5229	0.4348	-0.0881
Tabula Sapiens	skin	0.6165	0.6228	+0.0062	0.3701	0.3821	+0.0120
Tabula Sapiens	small intestine	0.7745	0.8041	+0.0296	0.5569	0.5855	+0.0286
Tabula Sapiens	tongue	0.4278	0.7491	+0.3214	0.4405	0.4367	-0.0038
Tabula Sapiens	vasculature	0.8826	0.8781	-0.0045	0.7081	0.6820	-0.0262
Colon cancer	GSE132465	0.5898	0.5946	+0.0048	0.4993	0.4806	-0.0187
Lung adenocarcinoma	GSE131907	0.3196	0.3455	+0.0260	0.3239	0.3393	+0.0153