TRACESEEK: SCALE-AWARE EVIDENCE SEEKING FOR PROGRESSION LEARNING FROM SINGLE-CELL POPULATION SNAPSHOTS

Anonymous authors
Paper under double-blind review

ABSTRACT

Disease processes are often observed through ordered population snapshots, where multiple progression explanations can remain compatible with the same data. We introduce TraceSeek, a framework for progression learning that, under a fixed Scientific Query Program, selects which atlas view to inspect and at which inferential unit a claim may be supported. TraceSeek couples a progression-claim layer with a typed analytical topology, a query-conditioned budgeted controller, and surrogate Marginal Utility Distillation (MUD), enabling evidence refinement, comparison, aggregation back to valid units, and abstention when evidence remains insufficient. In controlled evaluations across eleven disease atlases with a shared typed tool interface and a deterministic query compiler, TraceSeek improves scale-selection diagnostics, reduces invalid-unit and false-support proxy rates, and achieves higher mean family-specific Task Score than ReAct and other tool-using agents. On a 54-episode multi-atlas progression-chain suite, TraceSeek improves Task Score from 0.606 to 0.840 and path completeness from 0.500 to 0.917 relative to ReAct. Ablations show that topology constraints, active scale selection, and MUD contribute materially to these gains. An Alzheimer's disease case study further shows stable microglial early-response localization under leave-one-donor-out analysis and stage-associated DAM structure.

1 INTRODUCTION

Single-cell atlases encode development, disease progression, and treatment response across nested biological scales (Heumos et al., 2023; Tejada-Lapuerta et al., 2025; Bai et al., 2026). Progression claims, such as an early microglial response preceding DAM-like states, require multi-scale evidence that is compatible across donors, cell states, and gene programs. Fine views can recover rare biology that coarse summaries miss. Many conclusions remain valid only after aggregation back to independent donors (Zimmerman et al., 2021; Squair et al., 2021). Progression analysis therefore needs compatible biological units, a separation of technical and biological axes, and abstention when evidence is insufficient.

LLM-driven agents for single-cell analysis can chain tools effectively (Ding et al., 2025; Wang et al., 2025a; Xiao et al., 2026; Nouri et al., 2026). They are usually optimized for task completion or annotation accuracy on views prepared upstream. Without an explicit claim state, they may compare incommensurate objects, treat batch or replicate structure as time, and produce confident narratives under sparse support. Choosing an inappropriate scale among cell, niche, and patient yields fluent answers that fail to support a valid progression conclusion.

Figure 1 summarizes three common failure modes. Coarse summaries erase localized early responses. Fine analyses recover them yet may treat correlated cells as independent replicates (Zimmerman et al., 2021; Squair et al., 2021; Wagner & Kleiner, 2025). Unconstrained exploration consumes budget and increases false support under multiple testing (Nakatsuka et al., 2025). A reliable agent must control scope, scale, cost, and the inferential unit that may support a claim.

Agents that optimize reasoning or tool use typically operate in a fixed environment (Yao et al., 2023; Zhang et al., 2025; Ye et al., 2025b), including recent long-horizon variants (Lin et al., 2025; Wan et al., 2026). Active observation methods select informative frames or segments under context

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.

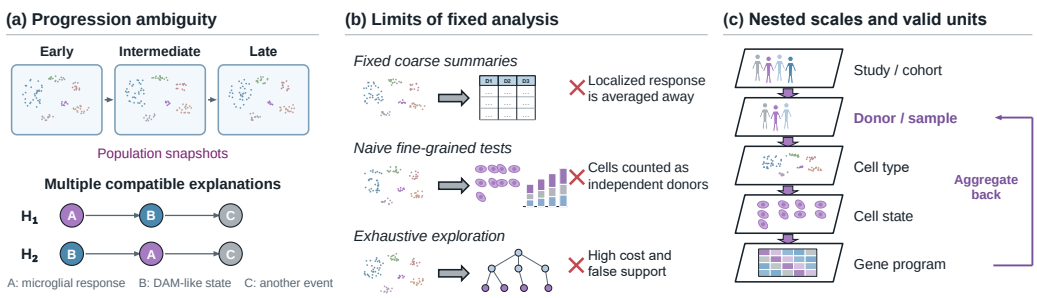


Figure 1: Problem formulation for progression learning from population snapshots. (a) Ordered views admit multiple explanations that disagree on event order. (b) Fixed summaries miss localized responses, and cell-level tests invite pseudoreplication. (c) Evidence must be refined across nested scales and aggregated to a legal inferential unit.

limits (Wang et al., 2025b; Li et al., 2026; Qiu et al., 2026), with validity judged mainly by answer accuracy. In single-cell settings, a predictive cell-level pattern cannot justify a donor-level progression conclusion on its own. Stronger embeddings and multi-sample models improve representations or sample-aware estimation (Cui et al., 2024; Boyeau et al., 2025; Ahlmann-Eltze & Huber, 2025). They still leave open which atlas view to inspect next and when a local finding must be aggregated back before a claim is issued.

We propose **TraceSeek** (Figure 2) for progression learning from ordered population snapshots. It couples two layers. The claim layer maintains a progression state over candidate ordered units and records support, conflict, exclude, and abstain decisions. Multiple explanations compatible with the observed snapshots can remain open until evidence resolves them. The evidence seeking engine comprises a typed analytical topology, a budgeted controller, multi-scale tools, and Marginal Utility Distillation (MUD). It searches for observations that update the claim state under cost constraints.

Our contributions are as follows.

- We formulate progression analysis on ordered single-cell snapshots as maintaining a compatible explanation set and seeking evidence that can support, conflict with, or force abstention on a claim.
- We present TraceSeek, which couples that claim interface with a typed topology, budgeted view and unit selection, and surrogate MUD under a fixed query contract.
- On eleven atlases under a shared tool interface, TraceSeek improves scale-selection diagnostics, unit-validity proxy metrics, and Task Score over strong agents; we further report disease/platform partitions, a small held-out MS/CRC suite, multi-atlas progression-chain scores, and an AD microglia and DAM case study.

All main tool-using-agent comparisons use a shared typed tool interface and a fixed SQP; main atlas evaluations use a deterministic query compiler to isolate evidence-routing performance.

2 Related Work

2.1 Language-model agents and active observation

Language-model agents combine reasoning with external actions, including tool-use loops, learned workflow construction, and step-value guidance (Yao et al., 2023; Zhang et al., 2025; Ye et al., 2025b; Lin et al., 2025; Wan et al., 2026). Scientific agents extend such orchestration to domain tools and databases (Ding et al., 2025; Wang et al., 2025a; Xiao et al., 2026; Nouri et al., 2026; Ghareeb et al., 2026). These systems improve planning over an available interface. The dataset view itself is usually fixed in advance, so which subset, biological scale, inferential unit, and fidelity to observe before tool execution often remains underspecified. We compare against general and single-cell agents that can be adapted to the shared typed action schema. Broader multi-agent systems such as CellAgent (Xiao et al., 2026) address related automation goals under a different interface contract.

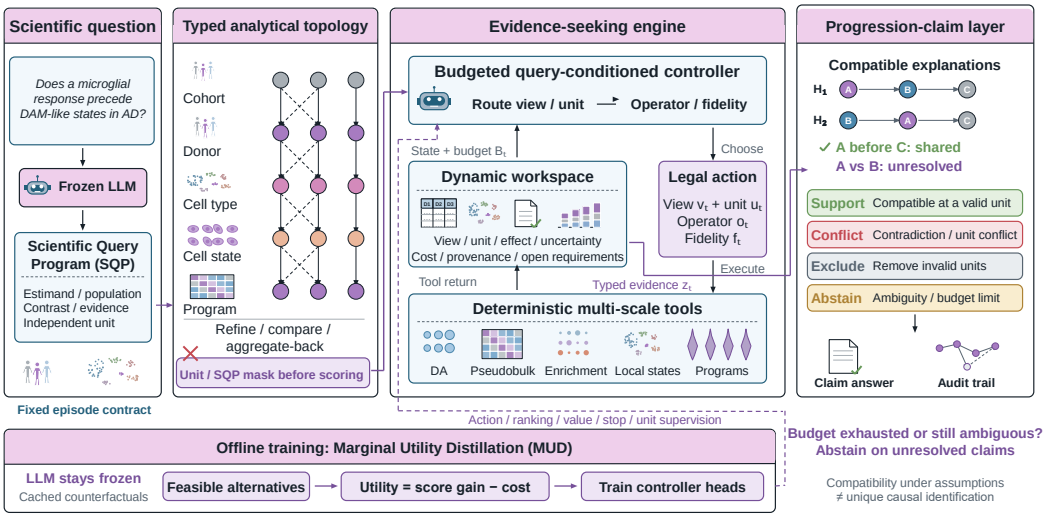


Figure 2: Overview of TraceSeek. A query compiler maps the question to an SQP over a typed topology. The controlled benchmark uses a deterministic compiler, and a frozen LLM is an optional natural-language interface. A budgeted controller selects legal actions, runs multi-scale tools, and updates a workspace. Structured returns feed a claim layer with support, conflict, and abstain decisions and an audit trail. MUD trains the controller offline.

Active observation methods select informative inputs under context and computation limits (Wang et al., 2025b; Buch et al., 2025; Li et al., 2026; Qiu et al., 2026; Qin et al., 2026; Pardyl et al., 2024). AdaGlimpse, for example, chooses glimpse position and scale in continuous visual exploration. Related stepwise value methods such as QLASS (Lin et al., 2025) supervise search with action utilities. Their observation units are typically frames, patches, or temporal segments, and validity is judged mainly by answer accuracy. Scientific atlases contain correlated subsamples and typed levels whose legal transitions depend on the estimand. Existing selectors therefore provide limited constraints on refinement, comparison, and aggregation back to a declared inferential unit, and they rarely maintain an explicit claim state that links observations to support, conflict, exclusion, or abstention.

2.2 Single-cell analysis, progression, and valid inference

Single-cell foundation models learn reusable cellular representations (Cui et al., 2024; Hao et al., 2024; Tejada-Lapuerta et al., 2025; Bai et al., 2026), and LLM systems automate annotation or standardized workflows (Hou & Ji, 2024; Crowley et al., 2025; Ye et al., 2025a; Yang et al., 2026; Xie et al., 2026; Nouri et al., 2026). These methods strengthen a prescribed representation or pipeline. They leave open how to answer questions for which the relevant population, resolution, and analysis path are initially unknown. Pseudotime and RNA velocity summarize continuous expression dynamics once a trajectory view is chosen (Heumos et al., 2023; Gayoso et al., 2024). TraceSeek asks whether a discrete claim about ordered units is supportable given multi-scale evidence, budget, and abstention.

Valid inference also requires separating cells as observations from donors as independent experimental units (Zimmerman et al., 2021; Squair et al., 2021). Pseudobulk, Milo, and multi-sample models provide sample-aware operators once the target view is known (Murphy & Skene, 2022; Dann et al., 2022; Boyeau et al., 2025; Ahlmann-Eltze & Huber, 2025). A remaining gap is a sequential mechanism that chooses when and where to invoke these operators, when a local finding must be aggregated back to the estimand unit, and how to revise an explicit claim state for progression before a statement is issued.

3 METHOD

We cast progression learning from ordered single-cell population snapshots as budgeted evidence seeking over a typed hierarchy with an explicit claim state (Figures 1–2). A query compiler maps the scientific question into an estimand. A typed analytical topology defines legal views and unit transitions. A query-conditioned controller selects the next observation under a remaining budget. Deterministic tools execute the chosen analysis. A compact workspace stores structured summaries. A claim layer revises support, conflict, exclusion, and abstention from those returns. MUD trains the controller from cached counterfactual observations. Raw expression matrices enter the language-model interface only through these structured summaries when that interface is enabled.

3.1 PROGRESSION CLAIMS

A progression claim is a tuple $(U, \preceq, \mathcal{E})$ where U is a set of biological units, $\preceq$ a proposed partial order or onset relation, and $\mathcal{E}$ the evidence so far. The claim layer maintains a small compatible set and records four decisions.

- **Support.** Evidence is compatible with $\preceq$ on a validated unit set.
- **Conflict.** Evidence contradicts $\preceq$ or mixes incompatible units.
- **Exclude.** Remove units that fail validity checks, including wrong scale or treating a technical replicate as stage.
- **Abstain.** The budget is exhausted or residual ambiguity remains above a threshold.

The layer records compatibility under declared assumptions. Reported support is restricted to that compatibility record. After each tool return, validated donor-level support strengthens compatible orders, illegal cell-level proposals trigger conflict on unit claims, contradictory contrasts mark conflict, and unresolved clauses yield abstention. The audit trail links every claim decision to the views and units that licensed it.

3.2 PROBLEM FORMULATION

Let a scientific dataset be represented by a typed analytical topology

$$\mathcal{H} = (\mathcal{V}, \mathcal{E}, \tau, \mu), \tag{1}$$

where each node $v \in \mathcal{V}$ is an analyzable view, $\tau(v)$ its biological scale, and $\mu(v)$ its metadata together with the independent experimental unit associated with that view. Edges in $\mathcal{E}$ encode legal relations among views, including refinement (for example cohort→cell type), conditioning, comparison across peer views, and aggregation back to a parent unit. In a single-cell atlas, a view may describe a donor cohort summary, a donor×cell-type pseudobulk matrix, a local cell-state partition, or a gene-program activity table.

The agent cannot inspect $\mathcal{H}$ exhaustively. At step t it observes a compact workspace b_t and remaining budget B_t, then selects a structured action

$$a_t = (v_t, u_t, o_t, f_t), \tag{2}$$

where v_t is the next view, u_t the inferential unit, o_t an analytical operator, and f_t a fidelity level. The environment executes a_t with deterministic tools and returns a structured observation z_{t+1} that may update the claim state. A trajectory $\tau = (a_1, \ldots, a_T)$ terminates by emitting an answer $\hat{y}$ together with an audit trail of views, units, costs, and claim decisions.

The learning objective balances answer quality against path cost and validity,

$$\max_{\pi} \ \mathbb{E}_{\pi}[S(\hat{y}, y^*) - \lambda C(\tau) - \rho V(\tau)] \,, \tag{3}$$

where S is the family-specific Task Score against grader y^*, C accumulates tool and data-access cost along the trajectory, and V penalizes invalid unit transitions. Correct unit choice is enforced in the environment so that cell-level observations are not treated as independent donor replicates for a donor-level estimand. A locally accurate intermediate result can still fail to support a claim when the observation unit is confused with the independent unit of the estimand.

3.3 QUERY PROGRAM AND ANALYTICAL TOPOLOGY

A query compiler converts the user question into a Scientific Query Program (SQP). The SQP specifies the estimand, target population, contrast, required evidence types, and the unit of inference at which the final claim must be supported. For example, a question about early disease-associated microglial activation is compiled to a donor-level association, even when the relevant molecular signal is localized within microglial states. The SQP is a frozen contract for the episode. Tools may refine scope within that contract and leave the declared inferential unit unchanged.

In the controlled atlas benchmark we use a deterministic compiler so that reported gains isolate evidence routing under a fixed contract. A frozen language model is an optional natural-language front end for the same SQP schema.

The SQP induces a typed mask over candidate actions. Let $m_q(a) \in \{0, 1\}$ indicate whether action a is compatible with the estimand and current workspace. Actions that treat cells as independent subjects for a donor-level contrast, use held-out labels to choose a test-time scope, or return a local effect without a required aggregate-back step are removed before policy scoring,

$$\mathcal{A}_t^q = \{a \in \mathcal{A}_t : m_q(a) = 1\}. \tag{4}$$

Topology edges define a unit-transition grammar. Legal transitions include zooming from donor summaries into a cell-type view, comparing peer cell types within the same donors, and aggregating a cell-state effect back to a donor-level score. Illegal transitions, including cell-level testing that replicates donor labels across cells, are excluded by construction. Statistical constraints are therefore encoded in the environment dynamics and need not be restated as prompts to the language model.

3.4 ACTIVE SCALE CONTROLLER

The controller receives the query embedding, the current frontier of $\mathcal{H}$, the workspace b_t, and the remaining budget B_t. Scoring the full Cartesian product of views and tools is intractable, so the policy is factorized as

$$\begin{aligned}\pi_\theta(a_t \mid q, b_t, B_t) = \pi_\theta(v_t, u_t \mid q, b_t) \\ \cdot \pi_\theta(o_t, f_t \mid v_t, u_t, q, b_t, B_t).\end{aligned} \tag{5}$$

The first factor routes the agent to a scientifically meaningful view and unit. The second chooses the operator and fidelity. Candidate nodes are encoded by scale, provenance, coverage, prior visitation, and residual query requirements. Query-conditioned message passing over the local neighborhood of the frontier estimates relevance and expected information gain under the remaining budget.

Scale changes use three transition classes. ***Refine*** moves toward a more localized state when the current view is too coarse for the estimand. ***Compare*** gathers evidence from a peer view at the same scale when contrast or specificity is unresolved. ***Aggregate*** returns a local finding to the inferential unit required by the SQP and thereby licenses claim updates at that unit. Because the next transition depends on the query and previous observations, the controller can skip unnecessary refinement, reverse course after a negative localization, or stop early when residual utility is non-positive.

3.5 DYNAMIC SCIENTIFIC WORKSPACE

Raw matrices and full tool logs stay outside the language-model context. Each observation is converted by a deterministic summarizer into a typed record

$$z_t = (\text{view}, \text{unit}, \text{effect}, \text{uncertainty}, \text{cost}, \text{provenance}). \tag{6}$$

The workspace b_t merges redundant observations, tracks which clauses of the SQP are already supported, and maintains an unresolved frontier of missing contrasts or missing aggregate-back steps. A readiness head $R_\psi(q, b_t)$ predicts whether the current evidence satisfies the answer contract. The agent stops when

$$R_\psi(q, b_t) > \eta \quad \text{and} \quad \max_{a \in \mathcal{A}_t^q} Q_\phi(a \mid q, b_t) < \epsilon, \tag{7}$$

that is, when coverage is sufficient and no remaining legal action has positive predicted utility. Stopping is therefore an explicit decision under the readiness and utility heads.

3.6 SURROGATE MARGINAL UTILITY DISTILLATION

Behavior cloning can reproduce demonstrated paths. Under a tight budget it provides little signal for why one observation is preferable to another. We therefore distill a *surrogate* marginal utility from cached counterfactual analyses. For a state b_t, the atlas trainer executes up to K feasible alternatives $\{a_t^k\}$ from the masked action set at a teacher trajectory state. Let $\Delta R(a_t^k)$ denote the change in workspace answer-readiness after executing a_t^k, and let b_{fam}, b_{unit}, and b_{teach} be small family, unit, and teacher-action bonuses recorded when building the cache. The surrogate utility target used for the main-table controller is

$$U_t(a_t^k) = \Delta R(a_t^k) + b_{\text{fam}}(a_t^k) + b_{\text{unit}}(a_t^k) + b_{\text{teach}}(a_t^k) - \lambda C(a_t^k), \tag{8}$$

with λ=0.05 on fidelity cost. This readiness-based surrogate is cheaper to cache than full trajectory rollouts of Equation 3 and is the target matched to the reported atlas tables. The same cache pipeline also supports TaskScore-aligned utilities when needed for ablations. Positive U_t identifies observations that raise readiness net of cost under that cached bonus scheme. Near-zero or negative utilities identify redundant refinement and expensive dead ends.

The controller is trained with value regression on U_t, pairwise ranking among candidates, and a readiness head,

$$\mathcal{L} = \mathcal{L}_{\text{value}} + \alpha\mathcal{L}_{\text{rank}} + \gamma\mathcal{L}_{\text{ready}}, \tag{9}$$

initialized from a behavior-cloning checkpoint. Targets come from deterministic tool outputs and the readiness and bonus utilities above. Main atlas evaluations keep the language-model front end disabled so that reported gains isolate controller routing under a fixed query contract.

The topology defines what may be observed, the controller selects the next observation, the workspace keeps evidence compact and auditable, MUD trains high marginal-utility selection under cost and validity constraints, and the claim layer records which progression statements those observations license.

4 EXPERIMENTS

We evaluate whether scale-aware, unit-constrained evidence seeking improves Task Score under a shared tool budget, and whether topology constraints and MUD drive those gains under budget and disease/platform partitions. All tool-using agents share the same typed action interface, tool library, budget B, and early-stop rule. Unless noted, point estimates are means over evaluation tasks. Overall Task Score means and selected controlled or transfer metrics report task-level 95% bootstrap CIs.

4.1 SETUP

Datasets. We evaluate on eleven disease atlases spanning scRNA-seq and snRNA-seq: IDC, ccRCC, PD_sc, AD, PD_sn, MS, CML, CRC, CC_sc, ODG, and PC. The overall agent comparison uses all eleven atlases under a frozen identifier split (99 queries). Controlled cross-scale and validity suites focus on AD, IDC, and ccRCC, where donor stage fields and nested cell-state structure are available for oracle-scale references (Mathys et al., 2024; Wu et al., 2021; Mei et al., 2024).

Baselines. Main comparisons use tool-using agents on the shared action schema $(\text{view}, \text{unit}, \text{operator}, \text{fidelity})$: ReAct (Yao et al., 2023), AFlow (Zhang et al., 2025), RaDAgent (Ye et al., 2025b), CellAtria (Nouri et al., 2026), and CASSIA (Xie et al., 2026), plus Heuristic and BC controllers that share the TraceSeek architecture without MUD. Foundation models and LLM annotators use fixed-view adapters and are reported separately as reference bands under a coarser interface. Baseline agents are evaluated through the common typed action interface rather than their native end-to-end configurations, ensuring matched tools, budgets, and action spaces.

Metrics and protocol. Primary metrics are the family-specific Task Score, unit accuracy, graded proxies for pseudoreplication, invalid aggregation, and false support, together with cross-scale agreement, oracle-distance diagnostics, and budget AUC. Task Score is computed by programmatic family-specific graders rather than an LLM judge; validity fields are graded proxies. Overall comparison uses B=10. Budget sweeps use $B \in \{3, 5, 10\}$ with seeds $\{0, 1\}$.

4.2 SCALE SELECTION AND INFERENTIAL VALIDITY

Table 1 reports mechanism recovery, autonomous scale choice, oracle-distance diagnostics, and scale accuracy on the controlled AD/IDC/ccRCC suite. TraceSeek attains strong scale diagnostics (mechanism 0.887, scale accuracy 0.867, and recorded oracle-distance 0.024), while fixed workflows and generic ReAct leave larger distances to the oracle-scale condition.

Table 2 compares unguarded unit accuracy and graded invalidity proxies when illegal unit actions remain available. TraceSeek-MUD reaches unit accuracy 0.931 with false-support proxy 0.064, and reduces pseudoreplication and invalid-aggregation proxies to 0.046 and 0.032, well below ReAct (0.765 unit accuracy and 0.300 false-support proxy).

Table 1: Cross-scale selection on AD/IDC/ccRCC (Gap: recorded Task Score distance to an oracle-scale reference condition).

Method	Mech.↑	Auto.↑	Gap↓	Scale↑	Task↑
Fixed Workflow	0.417	0.307	0.543	0.600	0.241
Fixed Coarse	0.623	0.594	0.256	0.617	0.325
Fixed Fine	0.743	0.772	0.078	0.602	0.676
Rule Coarse-to-Fine	0.712	0.683	0.167	0.706	0.755
ReAct	0.472	0.419	0.431	0.522	0.788
CASSIA	0.647	0.535	0.315	0.833	0.708
Heuristic	—	—	—	0.761	0.887
BC	—	—	—	0.900	0.928
TraceSeek	**0.887**	**0.874**	**0.024**	**0.867**	**0.928**

Table 2: Inferential-validity proxies with illegal unit actions available (Grd.: hard masking of illegal units at test time; FS and related fields are graded proxies).

Method	Unit↑	Pse.↓	Inv.↓	FS↓	Task↑	Grd.↑
Naive Pseudorep	0.259	0.729	0.729	0.687	0.295	0.471
ReAct	0.765	0.295	0.305	0.300	0.712	0.794
AFlow	0.832	0.220	0.140	0.244	0.814	0.902
RaDAgent	0.655	0.454	0.423	0.445	0.641	0.892
CellAtria	0.895	0.121	0.083	0.146	0.780	0.810
CASSIA	0.869	0.123	0.123	0.131	0.753	0.845
Heuristic	0.862	0.119	0.050	0.159	0.869	0.880
BC	0.731	0.145	0.145	0.152	0.753	0.850
TraceSeek-MUD	**0.931**	**0.046**	**0.032**	**0.064**	**0.939**	**0.947**

4.3 BUDGET, TRANSFER, AND ABLATIONS

Figure 3 shows Task Score versus tool budget with task-level 95% bootstrap intervals, and Task Score versus normalized cost. MUD is strongest at $B{=}3$ and remains competitive as the budget grows, while several agents plateau early or spend more normalized cost for similar Task Score. Table 3 reports transfer partitions with budget AUC, then recorded system ablations of the same released architecture. Aggregating over $B \in \{3, 5, 10\}$, TraceSeek-MUD attains budget AUC 0.924, with 0.909 on MS/CRC and 0.944 on PD_sn. A hold-out controller trained without MS/CRC matches Task Score 1.000 on a small MS/CRC hard subset. Because this small hold-out suite reaches ceiling, we interpret it as a stability check under disease exclusion rather than evidence that MS/CRC transfer is solved. Replacing MUD with BC yields the largest in-domain drop (0.257). Removing topology constraints yields the largest budget AUC drop (0.381), while removing inference constraints is weakest on the PD_sn platform partition. Removing active scale selection drops in-domain score by 0.130 and AUC by 0.311.

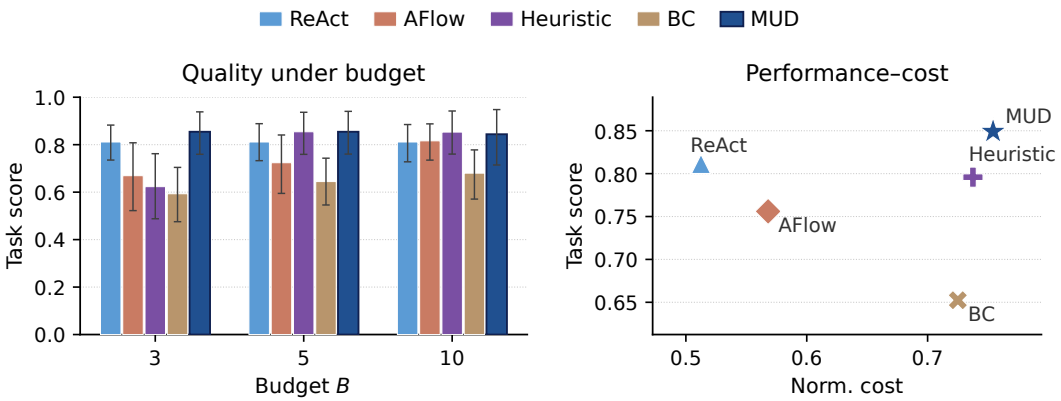


Figure 3: Budgeted navigation: Task Score versus tool budget with task-level 95% bootstrap intervals (left) and versus normalized cost at B=10 (right, point estimates).

Table 3: Transfer Task Score at B=10 and budget AUC over $B \in \{3, 5, 10\}$, with recorded system ablations of TraceSeek-MUD. MS/CRC and PD_sn are disease/platform partitions. A small MS/CRC controller hold-out subset and bootstrap CIs are reported in the appendix.

Policy / Variant	In-Dom.↑	MS/CRC↑	PD_sn↑	AUC↑
Transfer baselines				
Fixed	0.326	0.326	0.326	0.391
Rule	0.771	0.733	0.760	0.695
ReAct	0.808	0.784	0.771	0.759
AFlow	0.818	0.748	0.788	0.770
RaDAgent	0.759	0.738	0.741	0.795
CASSIA	0.750	0.719	0.725	0.696
Heuristic	0.836	0.843	0.875	0.845
BC	0.676	0.620	0.684	0.663
TraceSeek-MUD	**0.933**	**0.909**	**0.944**	**0.924**
Ablations vs. TraceSeek-MUD				
w/o Active Scale Selection	0.803	0.768	0.917	0.613
w/o Topology Constraints	0.772	0.761	0.550	0.543
w/o Inference Constraints	0.873	0.828	0.493	0.839
w/o Aggregate-Back	0.861	0.831	0.750	0.845

4.4 Overall Agent Comparison

Under budget B=10, TraceSeek attains mean Task Score 0.837 $[0.770, 0.896]$ among shared-interface agents (Table 4), exceeding ReAct at 0.788 $[0.742, 0.831]$ and remaining above 0.80 on every atlas. AFlow and RaDAgent are competitive on selected atlases but drop elsewhere, whereas TraceSeek stays consistently high.

Table 4: Overall Task Score on eleven atlases (B=10; Mean: 95% bootstrap CI). Shared-interface agents only. Fixed-view adapters are reported separately.

Category	Method	IDC	ccRCC	PD_sc	AD	PD_sn	MS	CML	CRC	CC_sc	ODG	PC	Mean [95% CI]
General Agent	ReAct	0.781	0.754	0.788	0.820	0.790	0.768	0.770	0.800	0.798	0.789	0.806	0.788 [.742,.831]
	AFlow	0.546	0.528	0.400	0.365	0.400	0.697	0.736	0.795	0.832	0.806	0.810	0.629 [.559,.697]
	RaDAgent	0.783	0.766	0.794	0.818	0.824	0.726	0.740	0.746	0.750	0.739	0.757	0.767 [.691,.837]
Single-cell Agent	CellAtria	0.725	0.702	0.735	0.764	0.759	0.571	0.626	0.615	0.639	0.618	0.633	0.672 [.617,.722]
	CASSIA	0.648	0.642	0.638	0.647	0.641	0.624	0.697	0.737	0.743	0.725	0.747	0.681 [.610,.749]
Ours	**TraceSeek**	**0.844**	**0.809**	**0.854**	**0.864**	**0.854**	**0.858**	**0.815**	**0.822**	**0.848**	**0.815**	**0.823**	**0.837 [.770,.896]**

4.5 PROGRESSION NAVIGATION AND CASE STUDY

We evaluate progression-facing queries in two complementary settings. Table 5 reports a multi-atlas disease-progression chain suite under the shared tool interface (n=54 episodes per method). TraceSeek-MUD attains Task Score 0.840 $[0.804, 0.872]$ with path completeness 0.917, exceeding ReAct (0.606 Task Score and 0.500 path completeness) and other shared-interface agents. On a matched-budget AD progression-navigation query ($B \in \{3, 5, 10\}$; Table A8), TraceSeek-MUD reaches Task Score 0.625, 0.625, and 0.812 and completes Path = 4/4 at B=10, exceeding ReAct and the heuristic on the evidence-path checklist (Path $2/4$).

On the same AD atlas, microglia localize without a pre-specified cell type (early-response score 0.24, top-1 in 36/48 leave-one-donor-out folds, and top-3 in 47/48). DAM program scores rise with donor stage (partial ρ=0.44 after age and sex adjustment, max-T adjusted p=0.008 over four programs). Figure 4 visualizes early-response ranking, DAM versus donor stage, and stage-wise marker programs. Path completeness makes the same point: TraceSeek-MUD reaches 0.917 versus 0.500 for ReAct on the chain suite, and Path = 4/4 versus $2/4$ on the AD query at B=10.

Table 5: Multi-atlas disease-progression chain suite under the shared tool interface (n=54 per method). Path completeness is the four-step required-chain hit rate used by the family grader.

Method	Task Score↑	Path completeness↑
TraceSeek-MUD	0.840 [0.804,0.872]	0.917 [0.884,0.949]
ReAct	0.606 [0.589,0.619]	0.500 [0.500,0.500]
AFlow	0.510 [0.433,0.588]	0.500 [0.444,0.556]
Heuristic	0.405 [0.352,0.455]	0.417 [0.384,0.449]
Rule	0.458 [0.389,0.526]	0.500 [0.444,0.556]
BC	0.131 [0.131,0.131]	0.250 [0.250,0.250]
CASSIA	0.103 [0.091,0.115]	0.167 [0.134,0.199]
RaDAgent	0.131 [0.131,0.131]	0.250 [0.250,0.250]
Fixed	0.131 [0.131,0.131]	0.250 [0.250,0.250]

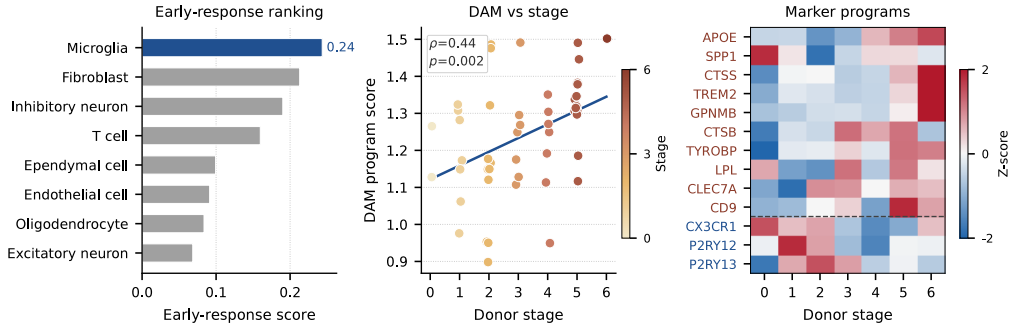


Figure 4: AD case study: early-response ranking (left), DAM versus donor stage (middle), and stage-wise marker programs (right).

5 CONCLUSION

We study progression analysis from ordered single-cell population snapshots, where agents must choose which atlas view and inferential unit to observe next, and which claims remain supportable under that evidence. TraceSeek couples a progression-claim interface with a typed, budgeted evidence-seeking engine trained by surrogate MUD under a fixed query contract. Under a shared typed tool interface and deterministic query compiler, it improves scale-selection diagnostics, reduces invalid-unit and false-support proxy rates, and raises mean Task Score over ReAct and other tool-using agents; on the multi-atlas progression-chain suite, path completeness rises from 0.500 to 0.917 relative to ReAct. Ablations attribute substantial performance gains to topology constraints, active scale selection, and MUD, while a small MS/CRC hold-out check and an AD microglia–DAM case study provide complementary evidence for transfer stability and progression-relevant biological recovery. Reliable agents for progression analysis should learn which evidence to observe next, at which unit a claim may be supported, and when to abstain.

AI USE STATEMENT

Generative AI tools were used to assist with manuscript drafting and language editing, literature discovery, and the organization and discussion of experimental analyses. AI tools were also used for technical assistance during research development where applicable. All methodological decisions, experiments, numerical results, scientific interpretations, and final manuscript content were reviewed and verified by the authors, who take full responsibility for the work.

ETHICS STATEMENT

We adhere to the ICLR Code of Ethics. All experiments use publicly available single-cell and single-nucleus RNA-seq disease atlases released in prior publications. We collected no new patient samples and did not conduct a human-subjects study for this work. TraceSeek is a research method for budgeted evidence seeking and claim auditing on ordered population snapshots, and is not intended for clinical decision making.

REPRODUCIBILITY STATEMENT

We will release code for the typed analytical topology, budgeted controller, surrogate MUD training, claim layer, evaluation harness, query inventory, and family-specific graders, together with configuration files and episode logs needed to reproduce the main tables and figures. Dataset details, evaluation protocol, Task Score definitions, and extended results are documented in the appendix. Where third-party atlases impose access constraints, we provide download pointers and preprocessing scripts so that results can be regenerated from the public releases.

REFERENCES

Constantin Ahlmann-Eltze and Wolfgang Huber. Analysis of multi-condition single-cell data with latent embedding multivariate regression. *Nature Genetics*, 57:659–667, 2025. doi: 10.1038/s41588-024-01996-0.

Ding Bai, Shentong Mo, Ruiyi Zhang, Yingtao Luo, Jiahao Gao, Jeremy Parker Yang, et al. scLong: A billion-parameter foundation model for capturing long-range gene context in single-cell transcriptomics. *Nature Communications*, 17:2380, 2026. doi: 10.1038/s41467-026-69102-y.

Pierre Boyeau, Justin Hong, Adam Gayoso, Martin Kim, Jose L. McFaline-Figueroa, Michael I. Jordan, et al. Deep generative modeling of sample-level heterogeneity in single-cell genomics. *Nature Methods*, 22:2264–2274, 2025. doi: 10.1038/s41592-025-02808-x.

Shyamal Buch, Arsha Nagrani, Anurag Arnab, and Cordelia Schmid. Flexible frame selection for efficient video reasoning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 29071–29082, 2025.

George Crowley, Tabula Sapiens Consortium, and Stephen R. Quake. Benchmarking cell type and gene set annotation by large language models with anndictionary. *Nature Communications*, 16: 9511, 2025. doi: 10.1038/s41467-025-64511-x.

Haotian Cui, Chloe Wang, Hassaan Maan, Kuan Pang, Fengning Luo, Nan Duan, Bo Wang, et al. scGPT: Toward building a foundation model for single-cell multi-omics using generative AI. *Nature Methods*, 21:1470–1480, 2024. doi: 10.1038/s41592-024-02201-0.

Emma Dann, Neil C. Henderson, Sarah A. Teichmann, Michael D. Morgan, John C. Marioni, et al. Differential abundance testing on single-cell data using k-nearest neighbor graphs. *Nature Biotechnology*, 40:245–253, 2022. doi: 10.1038/s41587-021-01033-z.

Keyan Ding, Jing Yu, Junjie Huang, Yuchen Yang, Qiang Zhang, and Huajun Chen. SciToolAgent: A knowledge-graph-driven scientific agent for multitool integration. *Nature Computational Science*, 5:962–972, 2025. doi: 10.1038/s43588-025-00849-y.

Adam Gayoso, Philipp Weiler, Mohammad Lotfollahi, Dominik Klein, Justin Hong, Aaron Streets, Fabian J. Theis, and Nir Yosef. Deep generative modeling of transcriptional dynamics for RNA velocity analysis in single cells. *Nature Methods*, 21:50–59, 2024. doi: 10.1038/s41592-023-01994-w.

Ali E. Ghareeb, Benjamin Chang, Ludovico Mitchener, Angela Yiu, Caralyn J. Szostkiewicz, Dmytro Shved, Gavin J. Gyimesi, Jon M. Laurent, Samantha M. Wright, Muhammed T. Razzak, Andrew D. White, et al. A multi-agent system for automating scientific discovery. *Nature*, 655:497–505, 2026. doi: 10.1038/s41586-026-10652-y.

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*, 21:1481–1491, 2024. doi: 10.1038/s41592-024-02305-7.

Lukas Heumos, Anna C. Schaar, Christopher Lance, Anastasia Litinetskaya, Felix Drost, Luke Zappia, et al. Best practices for single-cell analysis across modalities. *Nature Reviews Genetics*, 24:550–572, 2023. doi: 10.1038/s41576-023-00586-w.

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

Keliang Li, Yansong Li, Hongze Shen, Mengdi Liu, Hong Chang, and Shiguang Shan. LensWalk: Agentic video understanding by planning how you see in videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 19518–19528, 2026.

Zongyu Lin, Yao Tang, Xingcheng Yao, Da Yin, Ziniu Hu, Yizhou Sun, and Kai-Wei Chang. QLASS: Boosting language agent inference via q-guided stepwise search. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 37942–37958, 2025.

Hansruedi Mathys, Carles A. Boix, Leyla A. Akay, et al. Single-cell multiregion dissection of Alzheimer's disease. *Nature*, 632:858–868, 2024. doi: 10.1038/s41586-024-07606-7.

Shenglin Mei, Adele M. Alchahin, Ioanna Tsea, et al. Single-cell analysis of immune and stroma cell remodeling in clear cell renal cell carcinoma primary tumors and bone metastatic lesions. *Genome Medicine*, 16:1, 2024. doi: 10.1186/s13073-023-01272-6.

Alan E. Murphy and Nathan G. Skene. A balanced measure shows superior performance of pseudobulk methods in single-cell rna-sequencing analysis. *Nature Communications*, 13:7851, 2022. doi: 10.1038/s41467-022-35519-4.

Nathan Nakatsuka, Drew Adler, Longda Jiang, Austin Hartman, Evan Cheng, Eric Klann, and Rahul Satija. Improving reproducibility of differentially expressed genes in single-cell transcriptomic studies of neurodegenerative diseases through meta-analysis. *Nature Communications*, 16:7436, 2025. doi: 10.1038/s41467-025-62579-z.

Nima Nouri, Ronen Artzi, and Virginia Savova. An agentic ai framework for ingestion and standardization of single-cell rna-seq data analysis. *npj Artificial Intelligence*, 2:8, 2026. doi: 10.1038/s44387-025-00064-0.

Adam Pardyl, Michał Wronka, Maciej Wołczyk, Kamil Adamczewski, Tomasz Trzciński, and Bartosz Zieliński. AdaGlimpse: Active visual exploration with arbitrary glimpse position and scale. In *European Conference on Computer Vision*, 2024.

Yaxuan Qin, Hefei Li, Wenqi Mu, and Yancheng He. Efficient frame selection for long video understanding via reinforcement learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 16944–16953, 2026.

Jihao Qiu, Lingxi Xie, Xinyue Huo, Qi Tian, and Qixiang Ye. Longvideo-r1: Smart navigation for low-cost long video understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 40505–40515, 2026.

Jordan W. Squair, Matthieu Gautier, Claudia Kathe, Mark A. Anderson, Nicholas D. James, Thomas H. Hutson, et al. Confronting false discoveries in single-cell differential expression. *Nature Communications*, 12:5692, 2021. doi: 10.1038/s41467-021-25960-2.

Alejandro Tejada-Lapuerta, Anna C. Schaar, Robert Gutgesell, Giovanni Palla, Lennard Halle, Mariia Minaeva, et al. Nicheformer: A foundation model for single-cell and spatial omics. *Nature Methods*, 22:2525–2538, 2025. doi: 10.1038/s41592-025-02814-z.

Maggie R. Wagner and Manuel Kleiner. How thoughtful experimental design can empower biologists in the omics era. *Nature Communications*, 16:7263, 2025. doi: 10.1038/s41467-025-62616-x.

Guangya Wan, Mingyang Ling, Xiaoqi Ren, Rujun Han, Sheng Li, and Zizhao Zhang. COMPASS: Enhancing agent long-horizon reasoning with evolving context. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics*, pp. 3360–3380, 2026. doi: 10.18653/v1/2026.acl-long.152.

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*, 22:1677–1685, 2025a. doi: 10.1038/s41592-025-02748-6.

Ziyang Wang, Shoubin Yu, Elias Stengel-Eskin, Jaehong Yoon, Feng Cheng, Gedas Bertasius, and Mohit Bansal. VideoTree: Adaptive tree-based video representation for LLM reasoning on long videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3272–3283, 2025b.

Sunny Z. Wu, Ghamdan Al-Eryani, Daniel L. Roden, et al. A single-cell and spatially resolved atlas of human breast cancers. *Nature Genetics*, 53:1334–1347, 2021. doi: 10.1038/s41588-021-00911-1.

Yihang Xiao, Jinyi Liu, Yan Zheng, Shaoqing Jiao, Jianye Hao, Xiaohan Xie, Limingzhi, 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.

Elliot Xie, Lingxin Cheng, Jack Shireman, Yujia Cai, Jihua Liu, Chitrasen Mohanty, Mahua Dey, et al. CASSIA: A multi-agent large language model for automated and interpretable cell annotation. *Nature Communications*, 17:389, 2026. doi: 10.1038/s41467-025-67084-x. Published online December 2025.

Chen Yang, Xianyang Zhang, and Jun Chen. Large language model consensus substantially improves the cell type annotation accuracy for scrna-seq data. *Communications Biology*, 2026. doi: 10.1038/s42003-026-10420-8.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. ReAct: Synergizing reasoning and acting in language models. In *International Conference on Learning Representations*, 2023. URL `https://openreview.net/forum?id=WE_vluYUL-X`.

Wenjin Ye, Yuanchen Ma, Junkai Xiang, Hongjie Liang, Jintian Luo, Yuantao Li, Tao Wang, et al. Evaluation of cell type annotation reliability using a large language model-based identifier. *Communications Biology*, 8:1360, 2025a. doi: 10.1038/s42003-025-08745-x.

Yining Ye, Xin Cong, Shizuo Tian, Yujia Qin, Chong Liu, Yankai Lin, Zhiyuan Liu, and Maosong Sun. Rational decision-making agent with learning internal utility judgment. In *International Conference on Learning Representations*, 2025b.

Jiayi Zhang, Jinyu Xiang, Zhaoyang Yu, Fengwei Teng, Xiong-Hui Chen, Jiaqi Chen, Mingchen Zhuge, Xin Cheng, Sirui Hong, Jinlin Wang, Bingnan Zheng, Bang Liu, Yuyu Luo, and Chenglin Wu. AFlow: Automating agentic workflow generation. In *International Conference on Learning Representations*, 2025.

Kip D. Zimmerman, Mark A. Espeland, and Carl D. Langefeld. A practical solution to pseudoreplication bias in single-cell studies. *Nature Communications*, 12:738, 2021. doi: 10.1038/s41467-021-21038-1.

APPENDIX

Contents.

- Appendix A: Datasets and preprocessing
- Appendix B: Experimental protocol
- Appendix C: Task Score definition
- Appendix D: Implementation details
- Appendix E: Uncertainty summaries
- Appendix F: Additional experimental results

A DATASETS AND PREPROCESSING

This appendix documents the eleven disease atlases used throughout the paper. We treat stage fields as *ordered observational coordinates*, not as causal clocks.

A.1 SOURCE INVENTORY

Table A1 lists every source cohort contributing to an atlas. Controlled cross-scale and validity benchmarks center on AD, IDC, and ccRCC.

Table A1: Source cohorts for the eleven evaluation atlases.

Atlas	Modality	Source	N_cells	N_donors
IDC	scRNA-seq	GSE176078	70,651	20
	scRNA-seq	GSE195861	33,055	13
	scRNA-seq	GSE75688	549	11
ccRCC	scRNA-seq	GSE178481	150,551	26
	scRNA-seq	GSE210042	51,336	9
	scRNA-seq	GSE222703	27,963	6
	scRNA-seq	GSE224630	15,197	5
PD_sc	scRNA-seq	GSE223138	63,265	6
AD	snRNA-seq	syn52293442	1,033,482	48
PD_sn	snRNA-seq	GSE253462	265,135	41
	snRNA-seq	GSE243639	131,381	29
	snRNA-seq	GSE157783	41,428	11
MS	scRNA-seq	GSE179427	45,281	15
	scRNA-seq	GSE130119	22,077	9
CML	scRNA-seq	GSE218184	33,906	10
	scRNA-seq	GSE76312	1,553	26
CRC	scRNA-seq	GSE231559	39,949	12
CC_sc	scRNA-seq	E-MTAB-11948	40,723	5
	scRNA-seq	GSE224327	34,262	6
ODG	snRNA-seq	GSE205771	76,680	8
	snRNA-seq	GSE260928	8,793	2
	snRNA-seq	GSE260997	1,153	2
PC	scRNA-seq	GSE263733	55,384	21
	scRNA-seq	GSE197177	49,395	4
	scRNA-seq	GSE154778	13,216	16

A.2 STAGE SEMANTICS, METADATA, AND CLEANING

Common fields. Each atlas exposes a donor identifier, a cell-type / ontology label, optional batch/platform fields, and a primary ordinal stage field when a disease axis is available. Stage semantics follow the source cohorts (e.g., Braak for AD, AJCC-TNM for IDC/CRC/PC, WHO/ISUP

for ccRCC, HY for PD_sc, Braak_PD for PD_sn). Modality is scRNA-seq except AD, PD_sn, and ODG (snRNA-seq). Released dataset configurations record the exact field names.

Cleaning. Empty droplets and low-gene barcodes are removed with atlas-specific thresholds. Donor IDs are uniquified within each source before merging into an atlas view. When multiple sequencing platforms coexist, platform is retained as a typed attribute so that cross-platform transfer splits can hold out an entire platform.

Derived views. The topology exposes donor summaries, donor×cell-type pseudobulk matrices, local cell-state partitions, and gene-program activity tables. Program scores (e.g., DAM) are mean z-scored marker expression within the chosen unit. Pseudobulk sums counts within a donor×type group and applies library-size normalization before differential tests. Competence experiments subsample at most $8{,}000$ cells per run. Validity suites prefer donor-aggregated views whenever the SQP declares a donor-level estimand.

B Experimental Protocol

Due to space constraints in the main text, we summarize the evaluation suites, splits, baselines, and seeds here.

B.1 Query suite and splits

The hard-benchmark suite covers eleven atlases and seven task families. The families are statistical unit selection, composition versus expression, rare-state refinement, donor recurrence, batch-decoy rejection, cross-scale chains, and competence checks. Overall evaluation uses 99 dataset×task rows per method under budget $B{=}10$ with early stopping. The suite is balanced. Each atlas contributes nine hard queries, namely one each of annotation competence, batch-decoy rejection, clustering competence, cross-scale chain, donor recurrence, rare-state refinement, and statistical unit selection, plus two composition versus expression queries. Controlled cross-scale, validity, and budget suites center on AD, IDC, and ccRCC (27 task IDs, seeds $\{0, 1\}$).

Atlas identifiers and donor splits are frozen before controller tuning. Reference answers and Task Score rubrics follow the family-specific graders in Appendix C. A frozen query inventory records identifiers, atlases, families, and split tags for training, core evaluation, overall evaluation, controlled suites, and transfer partitions. Reported tables use these frozen identifiers under $B{=}10$.

MUD supervision for the main atlas controller uses readiness and bonus surrogate utilities on teacher trajectories from the training disease pool. The headline metric remains the family-specific evaluation Task Score. The released controller was trained on {IDC, ccRCC, PD_sc, MS, CRC}. A hold-out controller excludes MS/CRC from supervision and matches full Task Score on a held-out MS/CRC hard suite (Table A3).

B.2 Baselines and shared interface

All tool-using agents share the typed action interface (operator, unit, fidelity), the same tool library, budget B, and early-stop rule. Fixed-workflow baselines use a prespecified donor-aware view with sample-aware operators (coarse, fine, or coarse-to-fine) and serve as non-adaptive references under the shared tool budget. Foundation and LLM-annotator adapters use fixed coarse views with limited tool access (Table A7). Heuristic and BC share the TraceSeek controller architecture without MUD.

B.3 Seeds and uncertainty

Budget and transfer evaluations use seeds $\{0, 1\}$. Main-text overall Task Score means report task-level 95% percentile bootstrap CIs (2000 resamples over dataset×task scores). Budget curves use the same bootstrap within each $(method, B)$ cell. Bootstrap intervals for scale accuracy, transfer partitions, and ablations appear in Appendix E.

B.4 PROGRESSION-CHAIN AND CLAIM SUITES

Main-text Table 5 evaluates disease-progression chain queries from the transfer evaluation suite. There are seven unique chain queries (one each on CML, CRC, IDC, MS, ODG, PD_sn, ccRCC), each requiring the four-step sample→cell-type→refine→aggregate-back path under the shared interface. Per method, n=54 episodes equal evaluation partitions $\times$ budgets $B \in \{3, 5, 10\} \times$ seeds $\{0, 1\}$. The partitions are in-domain (IDC, ccRCC), unseen-disease (MS, CRC), cross-platform (PD_sn), and LODO (CML, CRC, MS, ODG), giving 9 disease-partition slots $\times$6 budget-seed runs. AD navigation (Table A8) uses a matched budget sweep of the AD progression-navigation query.

A controlled offline claim suite replays recorded tool sequences under a shared observation world model to stress-test the claim-update interface.

C TASK SCORE DEFINITION

Main-text Task Score columns are outputs of programmatic, family-specific graders (no LLM judge).

C.1 FAMILY-SPECIFIC GRADERS

The scoring rules used in the reported tables are

- **unit selection.** Unit-selection accuracy.
- **driver decomposition.** $0.6\,\text{claim} + 0.4\,\text{evidence-tool}$.
- **rare state.** $0.5\,\text{path} + 0.5\,\text{hit}$.
- **donor recurrence.** $0.4\,\text{path} + 0.6\,\text{recurrence}$.
- **batch aware.** $0.7\,\text{path} + 0.3\,\text{batch-mention}$.
- **cross-scale chain.** $0.75\,\text{chain-completeness} + 0.25\,\text{fine-label}$, with a penalty when completeness < 0.5.
- **clustering / annotation quality.** Quality of the corresponding tool return when that tool is present.

These formulas define the numbers in Tables 4–A8. They are graded recovery scores under an observational rubric. Several validity fields are themselves weighted or truncated proxies.

C.2 OPTIONAL THREE-COMPONENT AGGREGATE

For documentation, we also map path-completeness, result-quality, and unit-validity fields onto a common triple $(s_{\text{comp}}, s_{\text{scope}}, s_{\text{valid}})$:

$$\widetilde{\text{TaskScore}} = w_c\, s_{\text{comp}} + w_s\, s_{\text{scope}} + w_v\, s_{\text{valid}}, \tag{A1}$$

with default weights $(w_c, w_s, w_v) = (0.40, 0.30, 0.30)$. Equation A1 shows how the three proxies can be combined. The main tables use the family-specific graders above.

D IMPLEMENTATION DETAILS

D.1 ATLAS CONTROLLER

Main tables use the released atlas MUD controller. Table A2 lists its training recipe. Counterfactual utilities are readiness gains plus family, unit, and teacher bonuses minus a fixed cost coefficient (0.05). They are distinct from a replay of the family-specific Task Score graders. Training and evaluation use a deterministic query compiler and answer synthesizer with the language-model front end disabled, so that reported gains isolate controller routing under a fixed query contract.

A hold-out controller was trained on {IDC, ccRCC, PD_sc} only, excluding MS/CRC from supervision. On a 6-task MS/CRC hard suite at B=10, both the full and hold-out controllers reach mean Task Score 1.000 (Table A3), so the MS/CRC partition gains in the main text are preserved under disease hold-out. The same offline cache builder can optionally store grader ΔTaskScore utilities. Main tables use the readiness-based controller matched to the reported atlas results.

Table A2: Training hyperparameters for the atlas MUD controller used in the main tables.

Item	Setting
Training diseases	IDC, ccRCC, PD_sc, MS, CRC
Candidates per state K	up to 6 operators from the masked action set at the teacher state
Utility target	readiness gain + family/unit/teacher bonuses $-\ 0.05\times$ fidelity cost
Optimizer	Adam, learning rate 1×10^{-3}, weight decay 1×10^{-4}, 30 epochs (from BC init)
Loss terms	value MSE on chosen utility; pairwise hinge ranking among candidates (weight 0.15); readiness MSE (weight 0.3)
Language-model front end	disabled in main atlas evaluations (deterministic compiler/synthesizer)

Table A3: Hold-out transfer: MS/CRC Task Score at $B{=}10$ for the full atlas controller versus a hold-out controller trained without MS/CRC. Means over six hard tasks per disease.

Controller	Train diseases	MS	CRC
Full (main tables)	IDC, ccRCC, PD_sc, MS, CRC	1.000	1.000
Hold-out (no MS/CRC)	IDC, ccRCC, PD_sc	1.000	1.000

D.2 Controlled revision and language-model pilot

Two studies sit outside the main atlas engine and are reported separately from Tables 3 and 4.

First, a finite family ExtraTrees controller (6,021 candidate branches, $\lambda{=}0.04$, greedy continuation horizon ≤ 4) tests cost-aware utility selection under a locked claim family. Its hyperparameters and splits are distinct from the neural atlas MUD.

Second, a twelve-instance Qwen2.5-7B pilot with recorded prompts and responses yields six confirmations and zero false confirmations under budget 4, matching greedy on the same instances. That pilot documents that a live language-model front end can drive the same claim interface. The main atlas tables continue to use the deterministic compiler and planner.

D.3 Claim layer and compute

Claim updates consume structured tool returns (unit, operator, top early-response entities) and issue support, conflict, or abstain under fixed unit-consistency rules. Illegal cell or barcode proposals override prior donor-unit support and thereby block unsupported conversion of a donor-level claim into cell-level support. All reported engine tables are recomputed from frozen evaluation logs without retraining for this manuscript. The claim-decision suite replays recorded tool sequences offline with no additional atlas I/O.

E Uncertainty Summaries

Overall Task Score CIs appear in the main text. Here we report matching bootstrap 95% intervals for three controlled quantities that use smaller episode pools than the eleven-atlas overall comparison, namely scale accuracy on AD/IDC/ccRCC (Table A4), transfer Task Score partitions (Table A5), and in-domain ablations (Table A6). All intervals use percentile bootstrap with 2000 resamples at $B{=}10$ on the same episode pools as the point estimates in main-text Table 3.

On the controlled scale suite, TraceSeek-MUD attains mean scale accuracy 0.867 $[0.822, 0.911]$, between the oracle reference (0.944) and ReAct (0.522). Transfer intervals remain high for TraceSeek-MUD on in-domain, MS/CRC, and PD_sn partitions, consistent with the main-text transfer table. Ablation intervals show that removing topology constraints or active scale selection shifts the in-domain mean downward with non-overlapping or only weakly overlapping intervals relative to the full model, while removing aggregate-back yields a milder drop.

F Additional Experimental Results

This section collects adapter reference bands, a locked confirmation follow-up, auxiliary competence checks, and case-study stability analyses that complement the main-text agent comparison.

Table A4: Bootstrap 95% intervals for scale accuracy on the controlled AD/IDC/ccRCC suite.

Method	Mean	95% CI	n
Oracle	0.944	[0.907,0.981]	18
BC	0.900	[0.861,0.939]	18
TraceSeek-MUD	0.867	[0.822,0.911]	18
CASSIA	0.833	[0.722,0.944]	18
Heuristic	0.761	[0.684,0.844]	18
Rule Coarse-to-Fine	0.706	[0.664,0.751]	18
Fixed Workflow	0.600	[0.458,0.742]	18
ReAct	0.522	[0.369,0.687]	18

Table A5: Bootstrap 95% intervals for transfer Task Score partitions at $B{=}10$ (same episode pool as main-text Table 3; percentile bootstrap, 2000 resamples).

Setting	Method	Mean	95% CI	n
In-domain	**TraceSeek-MUD**	0.933	[0.897,0.966]	36
In-domain	ReAct	0.808	[0.743,0.873]	36
In-domain	Heuristic	0.836	[0.747,0.918]	36
In-domain	BC	0.676	[0.580,0.765]	36
MS/CRC	**TraceSeek-MUD**	0.909	[0.852,0.959]	36
MS/CRC	ReAct	0.784	[0.714,0.853]	36
MS/CRC	Heuristic	0.843	[0.746,0.927]	36
MS/CRC	BC	0.620	[0.512,0.726]	36
PD_sn	**TraceSeek-MUD**	0.944	[0.889,0.993]	18
PD_sn	ReAct	0.771	[0.676,0.876]	18
PD_sn	Heuristic	0.875	[0.791,0.958]	18
PD_sn	BC	0.684	[0.548,0.820]	18

Table A6: Bootstrap 95% intervals for in-domain Task Score under ablations at $B{=}10$. The full TraceSeek-MUD row uses the in-domain $B{=}10$ transfer episodes from Table 3; ablation variants use the recorded ablation episode pool.

Variant	Mean	95% CI	n
TraceSeek-MUD	0.933	[0.897,0.966]	36
w/o Inference Constraints	0.873	[0.797,0.942]	36
w/o Aggregate-Back	0.861	[0.802,0.919]	36
w/o Active Scale Selection	0.803	[0.731,0.873]	36
w/o Topology Constraints	0.772	[0.667,0.867]	36

F.1 Adapter baselines

Main-text Table 4 compares tool-using agents that share the typed action interface. Foundation models and LLM annotators operate under a coarser contract. They receive fixed coarse views with limited tool access and omit budgeted, unit-constrained evidence seeking. We therefore report them separately as reference bands under that coarser interface (Table A7).

On the same eleven-atlas suite at $B{=}10$, foundation adapters (scGPT, Nicheformer, scFoundation, scLong) reach mean Task Scores in roughly the 0.71–0.75 range. LLM annotators are more variable. mLLMCelltype and AnnDictionary approach 0.41 on average, while GPTCelltype and LICT fall lower, with particularly weak scores on several disease atlases. These bands sit below the shared-interface TraceSeek mean in the main text and show that fixed-view adapters leave substantial headroom on the progression evidence-seeking setting studied here.

F.2 AD progression-navigation budget sweep

Section 4.5 summarizes the matched-budget AD navigation query. Table A8 reports the full per-budget Task Scores and path checklist.

Table A7: Adapter baselines on the same eleven-atlas suite (B=10). Foundation and LLM-annotator methods use fixed coarse views with limited tool access; they are reported as reference bands, not peer agent comparisons.

Category	Method	IDC	ccRCC	PD_sc	AD	PD_sn	MS	CML	CRC	CC_sc	ODG	PC	Mean [95% CI]
Foundation Model	scGPT	0.753	0.761	0.776	0.759	0.728	0.728	0.717	0.743	0.750	0.741	0.759	0.747 [0.682,0.811]
	scFoundation	0.742	0.741	0.730	0.766	0.735	0.699	0.682	0.712	0.708	0.732	0.726	0.725 [0.655,0.794]
	Nicheformer	0.761	0.769	0.773	0.767	0.722	0.717	0.732	0.743	0.754	0.747	0.765	0.750 [0.683,0.813]
	scLong	0.724	0.716	0.713	0.747	0.728	0.662	0.666	0.697	0.698	0.714	0.707	0.707 [0.632,0.773]
LLM	GPTCelltype	0.320	0.309	0.311	0.318	0.312	0.170	0.145	0.193	0.170	0.204	0.178	0.239 [0.174,0.313]
	AnnDictionary	0.491	0.482	0.494	0.478	0.503	0.293	0.323	0.342	0.367	0.370	0.354	0.409 [0.335,0.487]
	LICT	0.421	0.416	0.412	0.421	0.415	0.170	0.115	0.208	0.170	0.193	0.170	0.283 [0.207,0.358]
	mLLMCelltype	0.696	0.666	0.707	0.729	0.739	0.185	0.115	0.163	0.155	0.170	0.170	0.409 [0.326,0.491]

Table A8: AD progression-navigation Task Score under matched budgets $B \in \{3, 5, 10\}$. Path is the four-step evidence-chain checklist on tool traces for the same query.

Method	B=3↑	B=5↑	B=10↑	Path
TraceSeek-MUD	0.625	0.625	0.812	4/4
Heuristic	0.219	0.625	0.625	2/4
BC	0.131	0.131	0.131	1/4
ReAct	0.625	0.625	0.625	2/4
Rule	0.131	0.625	0.812	3/4
CASSIA	0.038	0.131	0.131	1/4
Fixed	0.131	0.131	0.131	1/4

F.3 PROGRESSION-CLAIM EVALUATION

Table A8 reports Task Scores from the AD budget sweep on the progression-navigation query, together with a four-step evidence-path checklist from the same query's tool traces. Leave-one-donor-out frequencies in the AD/IDC case study (Appendix F.5) further check that the localized early-response population is stable under donor hold-out.

Separately, we run a finite family confirmation study that isolates cost-aware utility selection from the atlas neural controller. Scorers are trained and selected on locked splits, then evaluated on 512 held-out instances that are disjoint from earlier test data and from all training and validation data (Table A9). The revised configuration is chosen only on 256 new validation instances. Three estimator seeds share each test instance, and intervals cluster seeds within instances. Initial and revised scorers share the same inference module and are compared on identical observations.

At budget cap four, the revised utility selector reaches coverage 0.562 $[0.523, 0.600]$ versus greedy 0.510 $[0.467, 0.551]$, a paired correct-confirmation gain of $+0.052$ $[+0.014, +0.087]$, with near-zero confirmation error. Gains are largest at tight caps (2 and 4) and shrink once the budget is large enough that greedy already covers most confirmable instances (cap 8). This suite supports the claim that cost-aware utility selection improves confirmation under a matched response family. It leaves the main atlas MUD tables unchanged.

Table A9: Locked follow-up test: 512 new instances, disjoint from earlier test and all training/validation data. The revised configuration is selected only on 256 new validation instances. Three estimator seeds share each test instance. Intervals cluster seeds within instances. Initial and revised scorers share the same inference module and are compared on identical new observations.

Cap	Selector	Coverage [95% CI]	Error	Efficiency
2	Greedy	0.203 [0.170, 0.236]	0.000	0.118
2	Initial utility	0.210 [0.179, 0.240]	0.001	0.104
2	Revised utility	0.324 [0.285, 0.361]	0.000	0.170
4	Greedy	0.510 [0.467, 0.551]	0.002	0.160
4	Initial utility	0.486 [0.446, 0.525]	0.001	0.142
4	Revised utility	0.562 [0.523, 0.600]	0.002	0.165
8	Greedy	0.713 [0.674, 0.752]	0.002	0.150
8	Initial utility	0.712 [0.672, 0.751]	0.002	0.138
8	Revised utility	0.713 [0.674, 0.751]	0.002	0.146

F.4 AUXILIARY SINGLE-CELL COMPETENCE

Beyond agent Task Score, we also evaluate routine single-cell competence of the TraceSeek tool stack when used as a clustering and annotation backend. We measure clustering NMI, annotation macro-F1, and rare-state recall under a subsample of at most $8{,}000$ cells per run.

Table A10 recomputes AD and PD_sn under a corrected selection protocol with silhouette-based clustering selection and inner-validation F1 for annotation configuration search, applied uniformly to all compared methods (scVI omitted in this protocol). Donor held-out F1 and rare-state recall are reported alongside NMI. Under this protocol TraceSeek leads on AD NMI/F1/Rare and on PD_sn NMI, while remaining competitive on PD_sn donor F1 and rare recall (Harmony is strongest on those two PD_sn metrics). We treat these numbers as auxiliary evidence that scale-aware orchestration retains standard single-cell utility alongside the progression agent metrics in the main text.

Table A10: Competence under the corrected selection protocol (inner-validation F1 for annotation; silhouette for clustering; no scVI). Donor-held-out F1 / rare recall.

Dataset	Metric	PCA	Harmony	scVI	scGPT	TraceSeek
AD	NMI	0.713	0.715	—	0.730	**0.784**
	F1	0.916	0.916	—	0.912	**0.953**
	Rare	0.872	0.872	—	0.851	**0.912**
PD_sn	NMI	0.720	0.716	—	0.730	**0.791**
	F1	0.780	**0.879**	—	0.793	0.851
	Rare	0.389	**0.611**	—	0.389	0.574

F.5 CASE-STUDY LODO AND ADJUSTED AD ASSOCIATIONS

The main-text AD case study (Figure 4) reports that microglia localize as the early-response population and that DAM program scores rise with donor stage. Here we provide two stability checks behind that narrative.

Leave-one-donor-out ranking. Table A11 evaluates whether the top early-response population is stable when each donor is held out in turn. On AD, microglia remain top-1 in $36/48$ folds (0.750) and appear in the top-3 in $47/48$ folds (0.979). On IDC and ccRCC, the selected populations (CD8 T cell and renal epithelial, respectively) are top-1 in every LODO fold. These frequencies indicate that the localized ranking is not driven by a single donor outlier.

Age- and sex-adjusted program associations. Table A12 reports partial rank correlations between microglial program scores and donor stage, controlling for recorded numeric age and sex. The first block uses all 48 donors, with ages recorded as $90+$ encoded as 90. The second block excludes the 12 top-coded records (n=36). Intervals use donor bootstrap. Approximate HC3-studentized wild-bootstrap max-T p-values adjust over the four programs within each setting.

With all donors, DAM remains positively associated with stage (partial ρ=0.438 $[0.170, 0.675]$, adjusted p=0.008), while homeostatic and inflammatory programs are consistent with no association after adjustment. Lipid shows a weaker positive trend that loses significance after multiplicity adjustment. Excluding top-coded ages preserves a similar DAM point estimate (ρ=0.434) but widens the interval and raises the adjusted p to 0.079, so the association is sensitive to how extreme ages are coded. We therefore present both blocks and interpret DAM and stage coupling as supportive, age- and sex-adjusted observational evidence under the stated coding choices.

Table A11: Leave-one-donor-out early-response ranking stability on AD, IDC, and ccRCC.

Atlas	Top-1 population	LODO top-1	LODO in top-3
AD	Microglia	0.750	0.979
IDC	CD8 T cell	1.000	1.000
ccRCC	Renal epithelial	1.000	1.000

Table A12: AD age/sex sensitivity. First four rows: all 48 donors, recorded ages of 90+ encoded as 90. Last four: exclude the 12 top-coded records. Partial rank correlations control for recorded numeric age and sex; intervals use donor bootstrap. Approximate HC3-studentized wild-bootstrap max-T p-values adjust over four programs within each setting.

Program	Donors	Partial ρ [95% CI]	Adjusted p
DAM	48	0.438 [0.170, 0.675]	0.0080
Homeostatic	48	-0.123 [-0.393, 0.152]	0.8312
Inflammatory	48	-0.063 [-0.333, 0.235]	0.9816
Lipid	48	0.322 [0.045, 0.574]	0.1386
DAM	36	0.434 [0.045, 0.709]	0.0786
Homeostatic	36	-0.046 [-0.376, 0.308]	0.9970
Inflammatory	36	-0.110 [-0.406, 0.245]	0.8750
Lipid	36	0.260 [-0.170, 0.597]	0.4846