LEARNED ENTERPRISE DATA COMPREHENSION: COMPRESSION AND ROUTING FOR DATA AGENTS

Anonymous authors
Paper under double-blind review

ABSTRACT

Structured-data agents in enterprise settings must reason over complex data environments whose relevant evidence is distributed across schemas, relationships, policies, and recurring business roles. Modern agentic systems often address this burden through reusable markdown-style memory or skill files that preserve previously discovered information for later queries, reducing the need to rediscover the same structure repeatedly. This is useful, but it obscures a natural division of labor: agents are well suited to semantic reasoning, while learned systems are well suited to predicting and organizing recurring structure. We introduce *latent equivalence learning* to bridge this gap. The framework separates persistent task-relevant identities from their dataset-relative realizations. In our realization, supporting and opposing evidence shape support-realized Gaussian prototypes that learn how those identities are expressed in a particular data environment, while soft-membership profiles retain distinctions lost under a hard assignment. A separate learned query-prototype system represents recurring evidential requirements and maps them through a learned compatibility function into the same persistent identity structure. This identity-factorized, query-conditioned routing materializes the relevant dataset-specific evidence for downstream reasoning, allowing the agent to operate over an already organized evidential state rather than reconstructing cross-schema structure at every query. On the Data Agent Benchmark, spanning 54 queries across 12 heterogeneous datasets, our full implementation achieves 94.67% dataset-macro stratified Pass@1 over five complete trials and $258/270$ successful raw query attempts, compared with 55.51% for the benchmark's Claude Opus 4.6 reference agent, ranking first among 40 leaderboard entries at submission.

1 INTRODUCTION

As foundation models and agent frameworks become broadly accessible, organizations increasingly face a different bottleneck: not obtaining a capable model, but exposing the relevant internal evidence to it. This problem is especially relevant for data agents, which answer natural-language questions over heterogeneous organizational data. Consider a company whose two products are managed by different departments, recording customers under `buyer_id` in one system and `customer_id` in the other. To answer "Which customers use both products?", an agent must identify the shared customer-identifier role, verify which records refer to the same customer, and combine the relevant data. The goal is to answer the company-wide question without requiring the user to reconcile the departmental schemas first. Such requests combine evidence discovery, retrieval, and execution (Ma et al., 2026). We refer to the resulting systems challenge as the *three Cs*: *computation*, the work required to locate and use relevant evidence; *cost*, the training and inference resources consumed; and *context*, the evidence that can be exposed to a language model at once.

Agentic decomposition alone does not resolve these problems. Adding more agents, calls, or reasoning steps can increase the amount of exploration without improving the organization of the evidence being explored. Recent benchmarks show that heterogeneous data questions remain difficult for frontier agents and that increasing the interaction horizon alone does not reliably produce successful long-horizon data analysis (Ma et al., 2026; Xu et al., 2026). At the opposite extreme, assigning the entire burden to a monolithic learned predictor is not automatically better. High-dimensional feature spaces make variable selection and representation design increasingly consequential (Guyon & Elis-

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.

seeff, 2003; Fan & Li, 2007); large training and tuning pipelines impose substantial computational and energy costs (Strubell et al., 2019; Schwartz et al., 2020); and models deployed in nonstationary settings remain vulnerable to distribution shift, degraded uncertainty calibration, and interference during continual adaptation (Koh et al., 2021; Ovadia et al., 2019; Parisi et al., 2019).

We argue that these difficulties partly reflect a mismatch in the division of labor between learned systems and language-model agents. In enterprise data environments, language-model agents are commonly asked both to discover the latent organization of an unfamiliar data system and to reason over that organization. Conversely, task-specific learned models are often expected to encode the full semantics of every possible query, schema, and execution path. Neither allocation uses the respective system where it is strongest.

Our central thesis is that enterprise data agents should reason over learned dataset-relative realizations of persistent task-relevant identities, rather than rediscovering those identities from dataset-specific surface forms at every query. We introduce *latent equivalence learning* as the representation-learning framework for constructing dataset-relative realizations of persistent identities from supporting and opposing evidence, together with a *query-conditioned routing system* that arises naturally once the learned evidence is organized around those shared identities. The routing system consumes this learned identity structure rather than reconstructing it: recurring query requirements are mapped to the persistent identities they require, and those identities in turn expose their realizations and associated evidence in the current dataset. The framework itself does not prescribe a particular prototype family or evidence-construction mechanism; the Gaussian-prototype system studied here is one concrete realization.

Our earlier example of a business query spanning multiple departmental representations of a customer identifier exposes the distinction that motivates the framework. The recurring role *customer identifier* is a persistent identity: its task-relevant meaning can be reused even when one system records it as `buyer_id`, another as `customer_id`, and their surrounding schemas differ. The departmental columns, records, relations, and other observed structures are instead evidence objects from which the dataset-specific realization of that identity must be learned. In this sense, the business role is held fixed across the modeled environments, while its realization is latent: learning determines how observed objects express that role from the available supporting and opposing evidence.

This separation extends beyond a single company. Organizations in different industries may instantiate comparable business roles using different names, schemas, and local features. For a scoped deployment domain, we model these recurring roles by a finite inventory of task-relevant identities and learn how each dataset realizes them, rather than attempting open-world semantic equivalence. The representation is trained to absorb variation that is consistent with a shared identity while preserving distinctions that the available evidence indicates remain task-relevant. Where the same roles recur, the identity structure can be reused while dataset-relative realizations are learned locally; genuinely new roles require extending the modeled inventory.

We instantiate this framework with support-realized Gaussian prototypes (Snell et al., 2017; Fort, 2017). The evidence-side model learns the dataset-relative realizations of the persistent identities just described, while a separate learned query-prototype model represents recurring evidential requirements. A learned compatibility map converts those requirements into activations over the persistent identity coordinates. In the example, a request concerning customers who use both products activates customer-identifier and product-relation requirements; the current dataset realization then determines which departmental columns, records, and relations provide evidence for those identities. A bounded view of the supporting, opposing, and competing evidence passes to the semantic Arbiter and tool-equipped data agent for interpretation, verification of customer-level correspondence, and computation. Query requirements can therefore be reused across environments even though the evidence that realizes them changes.

We then retain the full soft profiles because evidence or queries with the same hard identity may differ in secondary memberships that change which competing evidence should be surfaced. Separately, the Gaussian evidence-side realization admits a local assignment-stability bound characterizing when perturbations of an embedding preserve its hard identity under a fixed prototype bank. Finally, we instantiate the framework in a data-agent harness and evaluate it on the Data Agent Benchmark (DAB), which tests agents across heterogeneous schemas and multi-step data ques-

tions (Ma et al., 2026). The evaluation tests whether adding the learned equivalence representation improves a strong agentic system under a fixed downstream reasoner.

Contributions. We formulate latent equivalence learning through persistent identities, dataset-relative realizations, and abstract supporting and opposing evidence. We instantiate it with Gaussian evidence prototypes and a separate query-prototype system connected by a learned compatibility map. We characterize the resulting identity-factorized routing and derive a conditional local assignment-stability certificate, including its connection to the existing signed-evidence losses. We evaluate the resulting representation-and-routing system in a constrained data-agent harness.

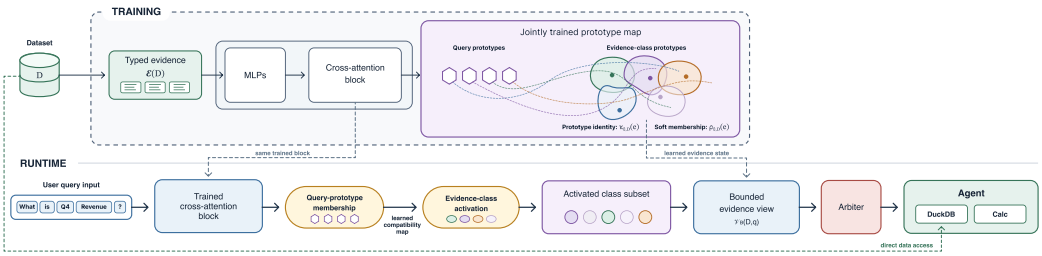

Figure 1: **Training and runtime of the proposed architecture.** Offline, heterogeneous evidence is encoded into dataset-relative prototype realizations of persistent identities, while a separate query-prototype system learns recurring evidential requirements and their compatibility with those identities. At runtime, a request activates persistent identity coordinates, the current dataset materializes a bounded view of their associated evidence, and the Arbiter and constrained data agent perform semantic interpretation and execution over that view.

2 Related Work

Invariance and prototype learning. Deep Sets builds permutation invariance into set representations, while large-margin metric learning learns distances that favor same-label neighbors over competing labels (Zaheer et al., 2017; Weinberger & Saul, 2009). Prototypical Networks represent episodic classes by support-derived centers; Gaussian Prototypical Networks extend this construction with precision-weighted support and class-dependent distances (Snell et al., 2017; Fort, 2017). We adopt these ingredients to realize persistent evidence identities for a particular dataset and connect them to a separate query-prototype system. Our local assignment certificate specializes the established relation between classification margins and Lipschitz sensitivity (Tsuzuku et al., 2018); the general certification principle is not claimed as a new contribution.

Structured correspondence and weak supervision. Sherlock and SATO infer semantic column types using column and table context, while Snorkel combines imperfect labeling sources into training supervision (Hulsebos et al., 2019; Zhang et al., 2020; Ratner et al., 2017). Classical structured-data methods provide complementary inductive biases: Lasso favors sparse linear support, while XGBoost captures nonlinear feature interactions through boosted trees (Tibshirani, 1996; Chen & Guestrin, 2016). These approaches illustrate how heterogeneous representations, predictors, and imperfect supervision can expose different evidence about the same structured environment. In our formulation, such outputs may contribute supporting or opposing evidence, but they do not themselves define the persistent identities. The distinction of interest is instead between a persistent identity and its dataset-relative realization, whose memberships and associated evidence remain available for subsequent query-conditioned use.

Evidence routing and data agents. Mixture-of-experts models route inputs to specialized computation, and retrieval-augmented generation conditions language-model generation on retrieved evidence (Jacobs et al., 1991; Shazeer et al., 2017; Lewis et al., 2020). StructRAG constructs task-appropriate representations of retrieved information before reasoning (Li et al., 2025). At the agent level, Data Interpreter uses hierarchical graph modeling and programmable node generation to organize and execute data-science workflows (Hong et al., 2024). Our focus is the learned interface

preceding agent execution: query-prototype memberships activate persistent evidence identities, and a dataset-specific map materializes the corresponding supporting and opposing evidence. This factorization distinguishes reusable query requirements from their realization in a particular data environment; the downstream agent retains responsibility for semantic interpretation and execution.

3 LATENT EQUIVALENCE LEARNING

We consider a scoped enterprise domain in which recurring task-relevant roles can be represented by a finite identity set shared across the modeled datasets. This separates what the framework holds fixed from what must adapt to each environment: the identity meanings provide the common reference structure, while learning determines how those identities are realized in a dataset, how observed evidence is assigned across them, and how a query activates them. We first formalize the evidence interface, then instantiate the dataset-relative realization with Gaussian prototypes, define the query-conditioned routing, and analyze assignment stability. Parameter dependence is suppressed once the trained model is fixed.

3.1 PROBLEM SETUP AND SIGNED EVIDENCE

The framework must distinguish the task-relevant roles that are shared across the modeled domain from the observed structures through which a particular dataset expresses them. Let D denote a structured data environment available to the system, and let $\mathcal{E}(D)$ be a finite set of *evidence objects* extracted from it. An evidence object $e \in \mathcal{E}(D)$ is an observed structured object—for example, a column, record group, or relation together with its available context—whose task-relevant role is to be represented by the model. Let $\mathcal{C}$ be a finite, nonempty set indexing the task-relevant roles covered by the modeled domain. We call these *persistent identities* because the meaning associated with each $c \in \mathcal{C}$ is shared across dataset realizations, even though the evidence that realizes it may change with D. Thus c denotes the shared role, whereas its dataset-relative realization will be learned from the evidence in $\mathcal{E}(D)$. To represent how an observed evidence object bears on a candidate identity, let

$$\omega_D^+, \omega_D^- : \mathcal{E}(D) \times \mathcal{C} \longrightarrow \mathbb{R}_{\geq 0}$$

For $(e, c) \in \mathcal{E}(D) \times \mathcal{C}$, $\omega_D^+(e, c)$ measures evidence supporting the assignment of evidence object e to identity c, while $\omega_D^-(e, c)$ measures evidence opposing that assignment. These maps form the interface between application-specific evidence construction and latent equivalence learning: the framework specifies how their outputs are used, but not how a particular application must produce them. In the Gaussian realization below, supporting weights contribute to prototype construction and attraction, whereas opposing weights penalize proximity to incompatible identity realizations. Thus supporting evidence determines which observed variation may be aligned, while opposing evidence preserves distinctions that remain task-relevant.

3.2 SOFT MEMBERSHIP AND ASSIGNMENT FIBERS

The signed evidence interface describes how individual observations support or oppose candidate identities; we next need a representation that records the resulting assignment structure without immediately collapsing it to a single label. Let $\Delta(\mathcal{C}) \stackrel{\text{def}}{=} \left\{ \pi \in \mathbb{R}_{\geq 0}^{|\mathcal{C}|} : \sum_{c \in \mathcal{C}} \pi_c = 1 \right\}$ be the probability simplex indexed by the persistent identities, and let $\rho_D : \mathcal{E}(D) \longrightarrow \Delta(\mathcal{C})$ assign each evidence object a normalized soft-membership profile. The coordinate $[\rho_D(e)]_c$ records the relative membership of evidence object e in identity c; Section 3.3 gives the learned Gaussian realization used in our experiments. To obtain a single identity when a hard readout is required, fix a deterministic selector

$$T : 2^{\mathcal{C}} \setminus \{\varnothing\} \longrightarrow \mathcal{C}, \quad T(A) \in A$$

for every nonempty $A \subseteq \mathcal{C}$. Define

$$\begin{aligned} \kappa_D(e) &\stackrel{\text{def}}{=} T\left(\arg\max_{c \in \mathcal{C}} [\rho_D(e)]_c \right), \\ e \sim_D e' \quad &\Longleftrightarrow \quad \kappa_D(e) = \kappa_D(e'), \qquad Q_{c,D} \stackrel{\text{def}}{=} \kappa_D^{-1}(\{c\}). \end{aligned} \tag{1}$$

Thus $\sim_D$ is an equivalence relation on $\mathcal{E}(D)$ induced by the hard identity readout, and its nonempty fibers partition the evidence objects. The identity c is the shared role label, whereas $Q_{c,D}$ is the dataset-dependent set of evidence objects assigned to that role. The hard fibers define equivalence classes, but they discard how strongly an object competes among identities. We therefore retain the full profile $\rho_D(e)$ so that later evidence materialization can distinguish objects with the same hard assignment but different secondary memberships. A natural comparison on these profiles is

$$d_D^{\mathrm{mem}}(e, e') \stackrel{\text{def}}{=} \|\rho_D(e) - \rho_D(e')\|_2.$$

This is a pseudometric: zero distance means identical membership profiles and therefore the same hard identity, while the converse need not hold. The Gaussian realization below learns ρ_D from prototype distances; the elementary quotient and pseudometric properties are verified in Appendix B.1.

3.3 Support-Realized Gaussian Prototypes

We now instantiate the assignment structure above with the metric realization used in our experiments. The purpose of this stage is to learn a geometry in which evidence supporting the same persistent identity is represented near a shared dataset-relative prototype, while evidence opposing that identity can be separated from it. This follows the support-based metric-learning view of Prototypical Networks (Snell et al., 2017, Section 2.2) and adopts precision-weighted prototype construction from Gaussian Prototypical Networks (Fort, 2017, Sections 3.1–3.3); latent equivalence learning itself does not require this particular realization. Let $m, p \in \mathbb{N}_{>0}$ denote the input-feature and learned metric dimensions. We use a feature map $\Phi_D : \mathcal{E}(D) \longrightarrow \mathbb{R}^m$ to encode each evidence object and its available context as the numerical input to the metric learner. This differs from $\omega_D^{\pm}(e, c)$: $\Phi_D(e)$ describes the evidence object itself, whereas $\omega_D^{\pm}(e, c)$ weight its association with a candidate identity. A learned encoder $\psi_\theta : \mathbb{R}^m \longrightarrow \mathbb{R}^p \times \mathbb{R}_{>0}$ produces $(z_e, s_e) = \psi_\theta(\Phi_D(e))$, where $z_e \in \mathbb{R}^p$ is the learned metric representation of e and $s_e > 0$ is its predicted scalar precision. For each $c \in \mathcal{C}$, let $\mu_c^0 \in \mathbb{R}^p$ and $s_c^0 \in \mathbb{R}_{>0}$ denote its learned prior center and scalar precision, and define $\Lambda_c^0 \stackrel{\text{def}}{=} s_c^0 I_p$. We now use the supporting weights to form the dataset-relative realization of identity c. Following the precision-weighted support aggregation of Gaussian Prototypical Networks (Fort, 2017, Section 3.3, Eqs. (5)–(6)), evidence with greater association weight and predicted precision contributes more strongly to the realized prototype. Define

$$s_{c,D}^{\mathrm{sup}} \stackrel{\text{def}}{=} \sum_{e \in \mathcal{E}(D)} \omega_D^{+}(e, c)\, s_e.$$

Whenever $s_{c,D}^{\mathrm{sup}} > 0$, the corresponding support-derived center is

$$\mu_{c,D}^{\mathrm{sup}} \stackrel{\text{def}}{=} \frac{\sum_{e \in \mathcal{E}(D)} \omega_D^{+}(e, c)\, s_e z_e}{s_{c,D}^{\mathrm{sup}}}.$$

Thus $s_{c,D}^{\mathrm{sup}}$ summarizes the total precision-weighted support for identity c, while $\mu_{c,D}^{\mathrm{sup}}$ is the corresponding support-derived center. Because the available dataset evidence may realize an identity with varying support, we retain the persistent prior as a fallback rather than replacing it unconditionally by the support-derived center. Let $\alpha_{c,D} \in [0, 1]$ control this interpolation when support is present: $\alpha_{c,D} = 1$ uses the dataset support entirely, while $\alpha_{c,D} = 0$ retains the persistent prior. The active prototype center is

$$\mu_{c,D} = \begin{cases} \alpha_{c,D}\mu_{c,D}^{\mathrm{sup}} + (1 - \alpha_{c,D})\mu_c^0, & s_{c,D}^{\mathrm{sup}} > 0, \\ \mu_c^0, & s_{c,D}^{\mathrm{sup}} = 0. \end{cases}$$

We apply the same support–prior interpolation to precision. For the scalar-precision realization used here, define

$$\Lambda_{c,D}^{\mathrm{sup}} \stackrel{\text{def}}{=} \big(\varepsilon + s_{c,D}^{\mathrm{sup}}\big) I_p, \quad \varepsilon > 0,$$

and let

$$\Lambda_{c,D} = \begin{cases} \alpha_{c,D}\Lambda_{c,D}^{\mathrm{sup}} + (1 - \alpha_{c,D})\Lambda_c^0, & s_{c,D}^{\mathrm{sup}} > 0, \\ \Lambda_c^0, & s_{c,D}^{\mathrm{sup}} = 0. \end{cases}$$

Since both component matrices are positive definite, so is $\Lambda_{c,D}$; the diagonal-precision case follows analogously by using coordinatewise positive precisions. The resulting dataset-relative realization

of identity c is $P_{c,D} \stackrel{\text{def}}{=} (\mu_{c,D}, \Lambda_{c,D})$. The three objects introduced so far play different roles: c is the shared identity label, $P_{c,D}$ is its learned geometric realization in dataset D, and $Q_{c,D} \subseteq \mathcal{E}(D)$ is the set of evidence objects assigned to that identity by the hard readout. To compare an embedded evidence object with a realized identity, define the precision-weighted prototype distance

$$d_{c,D} : \mathbb{R}^p \longrightarrow \mathbb{R}_{\geq 0}, \quad d_{c,D}(z) \stackrel{\text{def}}{=} \sqrt{(z - \mu_{c,D})^\top \Lambda_{c,D}(z - \mu_{c,D})}.$$

Following distance-based prototypical classification (Snell et al., 2017, Eq. (2)) and its precision-weighted Gaussian extension (Fort, 2017, Algorithm 1), we convert these competing distances into the soft-membership profile introduced in Section 3.2:

$$[\rho_D(e)]_c \stackrel{\text{def}}{=} \frac{\exp(-d_{c,D}(z_e)/\tau)}{\sum_{c' \in \mathcal{C}} \exp(-d_{c',D}(z_e)/\tau)}, \quad \tau > 0.$$

Thus $\rho_D(e)$ records relative membership across all realized identities, while $\kappa_D(e)$ selects an identity attaining the maximal membership, with ties resolved by T. We train this realization using the same supporting–opposing distinction that defines the evidence interface. Supporting evidence is encouraged to lie near the corresponding prototype,

$$\mathcal{L}_+ = \sum_{e \in \mathcal{E}(D)} \sum_{c \in \mathcal{C}} \omega_D^+(e, c)\, d_{c,D}^2(z_e),$$

whereas opposing evidence incurs a penalty whenever it lies within a prescribed squared-distance margin around the opposed identity realization,

$$\mathcal{L}_- = \sum_{e \in \mathcal{E}(D)} \sum_{c \in \mathcal{C}} \omega_D^-(e, c) \left[\xi_- - d_{c,D}^2(z_e)\right]_+^2, \quad \xi_- \in \mathbb{R}_{>0},$$

where $[x]_+ = \max\{x, 0\}$. Both terms are nonnegative: supporting evidence creates attraction toward the corresponding realization, while opposing evidence creates separation from inappropriate realizations. For a training episode, let $\mathcal{T}_{\text{qry}}$ denote its held-out evidence objects paired with target identities (e, y), where $y \in \mathcal{C}$. Following the episodic classification objective of Prototypical Networks (Snell et al., 2017, Section 2.2), define

$$\mathcal{L}_{\text{epi}} \stackrel{\text{def}}{=} - \sum_{(e,y) \in \mathcal{T}_{\text{qry}}} \log[\rho_D(e)]_y.$$

The complete training objective may then be written

$$\mathcal{L} = \mathcal{L}_{\text{epi}} + \lambda_{\text{soft}}\mathcal{L}_{\text{soft}} + \lambda_+\mathcal{L}_+ + \lambda_-\mathcal{L}_- + \lambda_{\text{unc}}\mathcal{L}_{\text{unc}} + \lambda_{\text{reg}}\mathcal{R},$$

with all coefficients nonnegative. Here $\mathcal{L}_{\text{soft}}$ preserves consistency with soft or weak targets, while $\mathcal{L}_{\text{unc}}$ and $\mathcal{R}$ collect realization-specific calibration and geometric regularization terms. The framework consumes application-specific evidence through $\omega_D^{\pm}$; their upstream construction remains separate from prototype aggregation. Having specified the learned geometry and the objective that shapes it, we can now characterize when its induced hard identity assignment is locally stable.

Local identity stability. For a fixed prototype bank, write $h_D(z) \stackrel{\text{def}}{=} T(\arg\min_{c \in \mathcal{C}} d_{c,D}(z))$, so that $\kappa_D(e) = h_D(z_e)$. The following margin certificate specializes the standard margin–Lipschitz argument to our prototype realization (Tsuzuku et al., 2018).

Proposition 1 (Local assignment stability)**.** *Fix D and its prototype bank, and suppose $|\mathcal{C}| \geq 2$. Let $e \in \mathcal{E}(D)$ have unique nearest identity $c_* = h_D(z_e)$. Set $L_{c,D} \stackrel{\text{def}}{=} \|\Lambda_{c,D}^{1/2}\|_{\text{op}}$, where $\|\cdot\|_{\text{op}}$ is the Euclidean operator norm, and define*

$$r_D(e) \stackrel{\text{def}}{=} \min_{c \neq c_*} \frac{d_{c,D}(z_e) - d_{c_*,D}(z_e)}{L_{c,D} + L_{c_*,D}} > 0. \tag{2}$$

For every $\delta \in \mathbb{R}^p$ with $\|\delta\|_2 < r_D(e)$, $h_D(z_e + \delta) = c_*$. Consequently, any $e' \in \mathcal{E}(D)$ with $\|z_{e'} - z_e\|_2 < r_D(e)$ belongs to $Q_{c_*,D}$.

Appendix B.3 proves the result and Appendix B.4 relates its margin to the supporting and opposing losses. The certificate concerns perturbations of the embedding with the bank fixed; arbitrary schema changes or rebuilt prototypes require additional control.

3.4 IDENTITY-FACTORIZED ROUTING AND AGENTIC COMPOSITION

The evidence-side representation tells us how the current dataset realizes the persistent identities; routing must determine which of those identities a request requires. We therefore represent each query independently in a second prototype space and then map its query-side memberships into activations over $\mathcal{C}$. Let $\mathcal{Q}$ denote the space of admissible natural-language requests, and let $(\mathcal{Z}_Q, d_Q)$ be a learned query metric space, with $d_Q : \mathcal{Z}_Q \times \mathcal{Z}_Q \longrightarrow \mathbb{R}_{\geq 0}$. A learned query encoder $g_\phi : \mathcal{Q} \longrightarrow \mathcal{Z}_Q$ maps a request q to $u_q \stackrel{\text{def}}{=} g_\phi(q)$. Let $\mathcal{K}$ index a finite, nonempty set of learned query prototypes $\{\eta_k\}_{k\in\mathcal{K}} \subseteq \mathcal{Z}_Q$, where each prototype represents a recurring evidential requirement. For $\tau_Q > 0$, define

$$[\beta(q)]_k \stackrel{\text{def}}{=} \frac{\exp(-d_Q(u_q, \eta_k)/\tau_Q)}{\sum_{\ell\in\mathcal{K}} \exp(-d_Q(u_q, \eta_\ell)/\tau_Q)}.$$

Thus $\beta : \mathcal{Q} \longrightarrow \Delta(\mathcal{K})$ assigns each request a soft profile over recurring query requirements. The query profile must next be expressed in the identity coordinates used by the evidence-side representation. Let $M \in \mathbb{R}^{|\mathcal{K}|\times|\mathcal{C}|}$ be a learned compatibility matrix, where M_{kc} represents the learned compatibility between query prototype k and persistent identity c. Define $\Gamma_M : \Delta(\mathcal{K}) \longrightarrow \Delta(\mathcal{C})$, where $\Gamma_M(b) \stackrel{\text{def}}{=} \mathrm{softmax}(M^\top b)$, and let

$$\gamma(q) \stackrel{\text{def}}{=} \Gamma_M(\beta(q)) = \mathrm{softmax}\big(M^\top \beta(q)\big)\,. \tag{3}$$

Thus $\gamma : \mathcal{Q} \longrightarrow \Delta(\mathcal{C})$ assigns each request an activation profile over persistent identities. The coordinates of $\rho_D(e)$ and $\gamma(q)$ share the index set $\mathcal{C}$, but they have different meanings: $\rho_D(e)$ represents evidence membership, whereas $\gamma(q)$ represents query demand. To reuse the learned dataset representation across requests, let S_D denote the finite, source-preserving evidence state produced for dataset D. Its records retain the learned embeddings and memberships together with supporting and opposing evidence, prototype diagnostics, and provenance required for later retrieval. A query should expose only a bounded subset of this state. Let $\mathrm{cost} : 2^{\mathsf{S}_D} \longrightarrow \mathbb{R}_{\geq 0}$ measure the context cost of a selected evidence view, and for a budget $B > 0$ define

$$\mathfrak{V}_B(D) \stackrel{\text{def}}{=} \{V \subseteq \mathsf{S}_D : \mathrm{cost}(V) \leq B\}\,.$$

We model query-conditioned materialization by $\mathsf{C}_B : 2^{\mathsf{S}_D} \times \Delta(\mathcal{C}) \longrightarrow \mathfrak{V}_B(D)$, and, for fixed D, define

$$\mathsf{F}_{D,B} : \Delta(\mathcal{C}) \longrightarrow \mathfrak{V}_B(D), \qquad \mathsf{F}_{D,B}(\pi) \stackrel{\text{def}}{=} \mathsf{C}_B(\mathsf{S}_D, \pi).$$

The bounded evidence view presented for request q is therefore

$$\mathcal{V}_B(D, q) \stackrel{\text{def}}{=} \mathsf{F}_{D,B}(\gamma(q)) = \big(\mathsf{F}_{D,B} \circ \Gamma_M \circ \beta\big)(q), \quad \mathcal{V}_B(D, q) \in \mathfrak{V}_B(D). \tag{4}$$

The materialization retains query-relevant supporting and opposing evidence, competing realizations, and provenance; source references permit scoped recovery of evidence omitted by the budget.

What the factorization preserves. Fix the identity inventory and query-side model, with no additional dataset-dependent inputs. Then $\gamma(q)$ depends only on the request: changing the dataset replaces $\mathsf{F}_{D,B}$ but leaves the query activation profile unchanged. For fixed D, budget B, and deterministic materialization,

$$\beta(q) = \beta(q') \implies \gamma(q) = \gamma(q') \implies \mathcal{V}_B(D, q) = \mathcal{V}_B(D, q').$$

Thus the factorization separates reusable query requirements from their dataset-specific evidence realization. It does not assert that evidence views are identical across datasets, that the requested evidence exists, or that the view determines the final answer. Appendix B.2 gives the corresponding formal statement and explains why a shared hard query-prototype label alone is insufficient. The bounded evidence view is then interpreted against the original request by a single LLM Arbiter. Let $\mathsf{R}_{\mathrm{Arb}}$ denote the space of typed, source-preserving Arbiter records. Define

$$\mathcal{A}_B(D, q) \stackrel{\text{def}}{=} \mathsf{Arb}(q, \mathcal{V}_B(D, q)) = (a_1, \ldots, a_{m_B}), \qquad a_j \in \mathsf{R}_{\mathrm{Arb}},$$

where $m_B < \infty$. Each record preserves its source reference and specifies evidence or execution requirements for the downstream agent. The Arbiter operates only on the bounded view; it does not search the full static state and therefore does not replace the learned routing stage. Let $\bigodot$

denote source-preserving composition of these records, with each component remaining individually accessible. The final answer is

$$\widehat{y} = \mathsf{Agent}\left(q, \bigodot_{j=1}^{m_B} a_j \mathsf{DuckDB}(D), \mathsf{Calc}\right).$$

The agent uses the routed records to retrieve required values, execute computations, and verify its response. Because both the Arbiter and the final agent retain the original request q, identical evidence views need not produce identical answers for different queries.

4 Experiments

4.1 Experimental Setup

Benchmark and evaluation protocol. We evaluate on the Data Agent Benchmark (DAB), an end-to-end benchmark for data agents spanning 54 queries across 12 datasets, nine application domains, and four database systems Ma et al. (2026). DAB includes heterogeneous structured-data tasks requiring operations such as multi-table reasoning, irregular join resolution, aggregation, temporal reasoning, and unstructured-text interpretation. One complete trial consists of one independent attempt at each of the 54 benchmark queries. We evaluate each primary Permute system over $n = 5$ complete trials, yielding $54 \times 5 = 270$ query attempts per system. Each query is evaluated by a query-specific validator. Let $\mathcal{D}$ denote the set of benchmark datasets, let $\mathcal{Q}_d \subseteq \mathcal{Q}$ denote the benchmark queries associated with dataset $d \in \mathcal{D}$, and let $r_{dqi} \in \{0, 1\}$ indicate whether trial i succeeds on query $q \in \mathcal{Q}_d$ in dataset d. The empirical success rate of a query is $\widehat{p}_{dq} = \frac{1}{n}\sum_{i=1}^{n} r_{dqi}$. DAB's primary metric is dataset-macro stratified Pass@1,

$$\mathrm{Pass@1}_{\mathrm{strat}} = \frac{1}{|\mathcal{D}|}\sum_{d\in\mathcal{D}}\frac{1}{|\mathcal{Q}_d|}\sum_{q\in\mathcal{Q}_d}\widehat{p}_{dq}.$$

Thus, queries are first averaged within each dataset and each dataset receives equal weight. The small markers in Figure 2 show the corresponding dataset-macro score for each complete trial, while the large markers show system means. We additionally report raw query-attempt accuracy,

$$A^{\mathrm{micro}} = \frac{\sum_{d\in\mathcal{D}}\sum_{q\in\mathcal{Q}_d}\sum_{i=1}^{n} r_{dqi}}{n\sum_{d\in\mathcal{D}}|\mathcal{Q}_d|}.$$

which weights each of the 270 query attempts equally rather than weighting datasets equally.

Systems and ablations. We compare three primary systems: the DAB Claude Opus 4.6 reference agent using the standard benchmark scaffold; Core, using the agentic harness with an Opus 5 reasoner but without the proposed learned equivalence representations; and our architecture, which adds the prototype realizations and query-conditioned evidence construction of Section 3. While DAB's Claude Opus 4.6 reference agent utilizes a weaker model, it serves as a baseline to highlight the stark differences performance can make, even when using a model with a relevant level of capability. The Core comparison retains the stronger agentic setting while omitting the learned representation. Furthermore, the broader DAB leaderboard provides additional system-level context for these comparisons.

4.2 Results and Ablations

Overall DAB performance. The DAB Claude Opus 4.6 reference agent obtains 55.51% dataset-macro stratified Pass@1, while Core reaches 84.13%. Our architecture reaches 94.67%, improving on Core by 10.54 percentage points and on the reference agent by 39.16 points. At the raw query-attempt level, our architecture succeeds on $258/270$ attempts (95.56%), compared with $236/270$ (87.41%) for Core, a net gain of 22 successes.

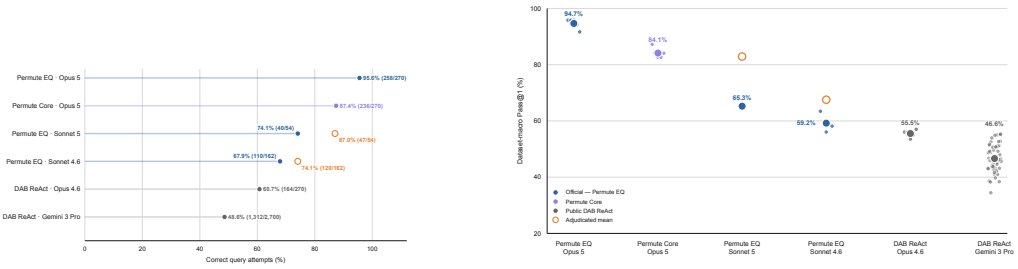

Figure 2: DAB performance by raw query-attempt accuracy (left) and dataset-macro stratified Pass@1 (right). Our proposed architecture is labeled PERMUTE EQ in the benchmark submission and figures, while PERMUTE CORE denotes the corresponding implementation using the conventional data-agent approach without the learned equivalence representation and identity-factorized routing.

Relative to Core, the gain is therefore obtained on top of an already strong agentic system; the distinguishing contribution is the learned identity-based representation and query-conditioned evidence routing upstream of the downstream reasoner. Comprehensive breakdowns of the query attempts and some additional relevant tracking metrics are detailed in the accompanying appendices.

5 CONCLUSION

Our results indicate that data-agent performance depends materially on how evidence is organized before it reaches the language model. In heterogeneous data environments, additional search and agentic decomposition can expand exploration without actually resolving the underlying problem of which evidence should be compared, preserved, or exposed to the reasoner. Latent equivalence learning moves this organization upstream by representing evidence through persistent task-relevant identities and materializing a bounded, query-conditioned view of their supporting, opposing, competing, and provenance-bearing evidence.

Taken together, these results support the practical value of separating reusable query requirements from dataset-specific evidence realization. DAB stresses this separation across heterogeneous schemas and query forms, including join resolution, aggregation, temporal reasoning, and unstructured-text interpretation, so the observed improvement is not confined to a single operation or data layout. The formal analysis gives a complementary view of the same behavior: under the stated query-side assumptions, the requirements induced by a request can remain fixed while their materialized evidence changes with the dataset, while within a fixed realization the hard evidence assignments are stable inside an explicit local margin. These are distinct forms of controlled reuse—across environments on the routing side and under perturbation on the representation side—and together explain how learned structure can be reused without requiring the downstream agent to reconstruct it for every request. The same analysis also makes clear where that claim stops.

That scope is finite. The framework assumes that a bounded inventory of persistent identities provides meaningful coverage of the deployment domain and that the available evidence is sufficient to realize and distinguish them. Requests that depend on identities outside this inventory, unsupported relationships, or distinctions absent from the evidence may therefore fall outside the modeled structure. Likewise, the local stability certificate is conditional on a fixed prototype bank; substantial changes to the learned realization require separate control. Performance consequently remains limited by identity coverage, evidence quality, routing, and the materialization budget.

Within these bounds, the broader implication is that evidence organization need not remain an implicit burden of the language model. Across the heterogeneous DAB tasks studied here, our results show that learning reusable structure upstream can improve data-agent performance while preserving a clear separation between representation and query-specific reasoning. This supports a broader design principle for structured-data agents: learn the organization that can be reused, and reserve language-model reasoning for the semantic interpretation and execution that remain specific to the request.

5.1 AI USE STATEMENT

AI tools, specifically ChatGPT 5.6 SOL Pro, were used to assist with the following:

- drafting and refining prose across sections;
- improving the formalization, consistency, and canonicalization of notation;
- assisting with code development, debugging, and ablation design; and
- assisting with the presentation of graphs, figures, and diagrams.

All mathematical proofs, implementation details, algorithmic constructions, code design, diagram design, and architectural decisions are the authors' own, although AI did help confirm any core questions surrounding some of these topics (in a limited scope). For any figures or images produced with AI assistance, the underlying content, structure, and intended presentation were designed by the authors prior to using AI; AI was used only to help render or refine those designs in a visually polished form. Similarly, AI was used extensively for proofreading and stylistic refinement of the manuscript, after which the authors independently reviewed, revised, and incorporated the resulting prose into the paper in their own style.

5.2 REPRODUCIBILITY STATEMENT

We provide the architectural definitions, training objectives, and inference procedure necessary to characterize the proposed method in the main text, with complete mathematical proofs and additional derivations provided in the appendix. The experimental section specifies the benchmark protocol, evaluation metrics, comparison systems, and repeated-trial methodology used to obtain the reported results. Certain implementation details, including hyperparameter configurations, are not disclosed, and the source code is not released because the system is currently proprietary.

All Data Agent Benchmark (DAB) evaluations strictly followed the official benchmark rules (cf. Data Agent Benchmark; Ma et al., 2026), and the reported results were externally validated. The validation protocol prohibited access to gold-standard answers and prohibited benchmark-specific information or leading cues from being incorporated into the system inputs or prompts. Consequently, the system was required to recover the information necessary to answer each query through its learned representations, data-agent execution, and subsequent arbiter reasoning rather than through access to benchmark answers or benchmark-specific prompt tuning.

Compute resources. All training and evaluation were conducted locally on two consumer-grade workstations; no external compute clusters or cloud-based training infrastructure were used. The first workstation was equipped with an NVIDIA GeForce RTX 3070 Ti with 8 GB of GPU memory, an Intel Core i9-9900K CPU, and 32 GB of system memory. The second workstation was equipped with an NVIDIA GeForce RTX 4060 Ti with 16 GB of GPU memory, an Intel Core i9-9900K CPU, and 128 GB of system memory. Both systems had at least 1,TB of local storage available throughout the experiments.

REFERENCES

Tianqi Chen and Carlos Guestrin. XGBoost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 785–794, 2016. doi: 10.1145/2939672.2939785.

Jianqing Fan and Runze Li. Statistical challenges with high dimensionality: Feature selection in knowledge discovery. In *Proceedings of the International Congress of Mathematicians, Madrid, August 22–30, 2006*, volume 3, pp. 595–622. European Mathematical Society, 2007. doi: 10.4171/022-3/31.

Stanislav Fort. Gaussian prototypical networks for few-shot learning on Omniglot. *arXiv preprint arXiv:1708.02735*, 2017. URL https://arxiv.org/abs/1708.02735.

Isabelle Guyon and André Elisseeff. An introduction to variable and feature selection. *Journal of Machine Learning Research*, 3:1157–1182, 2003.

Sirui Hong, Yizhang Lin, Bang Liu, Bangbang Liu, Binhao Wu, Ceyao Zhang, Chenxing Wei, Danyang Li, Jiaqi Chen, Jiayi Zhang, Jinlin Wang, Li Zhang, Lingyao Zhang, Min Yang, Mingchen Zhuge, Taicheng Guo, Tuo Zhou, Wei Tao, Xiangru Tang, Xiangtao Lu, Xiawu Zheng, Xinbing Liang, Yaying Fei, Yuheng Cheng, Zhibin Gou, Zongze Xu, and Chenglin Wu. Data interpreter: An LLM agent for data science, 2024. URL https://arxiv.org/abs/2402.18679.

Madelon Hulsebos, Kevin Hu, Michiel Bakker, Emanuel Zgraggen, Arvind Satyanarayan, Tim Kraska, Çağatay Demiralp, and César Hidalgo. Sherlock: A deep learning approach to semantic data type detection. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2019. doi: 10.1145/3292500.3330993. URL https://vis.csail.mit.edu/pubs/sherlock.

Robert A. Jacobs, Michael I. Jordan, Steven J. Nowlan, and Geoffrey E. Hinton. Adaptive mixtures of local experts. *Neural Computation*, 3(1):79–87, 1991.

Pang Wei Koh, Shiori Sagawa, Henrik Marklund, Sang Michael Xie, Marvin Zhang, Akshay Balsubramani, Weihua Hu, Michihiro Yasunaga, Richard Lanas Phillips, Irena Gao, Tony Lee, Etienne David, Ian Stavness, Wei Guo, Berton Earnshaw, Imran Haque, Sara Beery, Jure Leskovec, Anshul Kundaje, Emma Pierson, Sergey Levine, Chelsea Finn, and Percy Liang. WILDS: A benchmark of in-the-wild distribution shifts. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 5637–5664. PMLR, 2021.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. Retrieval-augmented generation for knowledge-intensive nlp tasks. In *Advances in Neural Information Processing Systems*, volume 33, 2020. URL https://proceedings.neurips.cc/paper/2020/hash/6b493230205f780e1bc26945df7481e5-Abstract.html.

Zhuoqun Li, Xuanang Chen, Haiyang Yu, Hongyu Lin, Yaojie Lu, Qiaoyu Tang, Fei Huang, Xianpei Han, Le Sun, and Yongbin Li. StructRAG: Boosting knowledge intensive reasoning of LLMs via inference-time hybrid information structurization. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=GhexuBLxbO.

Ruiying Ma, Shreya Shankar, Ruiqi Chen, Yiming Lin, Sepanta Zeighami, Rajoshi Ghosh, Abhinav Gupta, Anushrut Gupta, Tanmai Gopal, and Aditya G. Parameswaran. Can AI agents answer your data questions? a benchmark for data agents. *arXiv preprint arXiv:2603.20576*, 2026. URL https://arxiv.org/abs/2603.20576.

Yaniv Ovadia, Emily Fertig, Jie Ren, Zachary Nado, D. Sculley, Sebastian Nowozin, Joshua V. Dillon, Balaji Lakshminarayanan, and Jasper Snoek. Can you trust your model's uncertainty? evaluating predictive uncertainty under dataset shift. In *Advances in Neural Information Processing Systems*, volume 32, pp. 13991–14002, 2019.

German I. Parisi, Ronald Kemker, Jose L. Part, Christopher Kanan, and Stefan Wermter. Continual lifelong learning with neural networks: A review. *Neural Networks*, 113:54–71, 2019. doi: 10.1016/j.neunet.2019.01.012.

Alexander Ratner, Stephen H. Bach, Henry Ehrenberg, Jason Fries, Sen Wu, and Christopher Ré. Snorkel: rapid training data creation with weak supervision. *Proceedings of the VLDB Endowment*, 11(3):269–282, November 2017. ISSN 2150-8097. doi: 10.14778/3157794.3157797. URL http://dx.doi.org/10.14778/3157794.3157797.

Roy Schwartz, Jesse Dodge, Noah A. Smith, and Oren Etzioni. Green AI. *Communications of the ACM*, 63(12):54–63, 2020. doi: 10.1145/3381831.

Noam Shazeer, Azalia Mirhoseini, Krzysztof Maziarz, Andy Davis, Quoc V. Le, Geoffrey E. Hinton, and Jeff Dean. Outrageously large neural networks: The sparsely-gated mixture-of-experts layer. *arXiv preprint arXiv:1701.06538*, 2017.

Jake Snell, Kevin Swersky, and Richard S. Zemel. Prototypical networks for few-shot learning. In *Advances in Neural Information Processing Systems*, volume 30, 2017.

Emma Strubell, Ananya Ganesh, and Andrew McCallum. Energy and policy considerations for deep learning in NLP. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 3645–3650. Association for Computational Linguistics, 2019. doi: 10.18653/v1/P19-1355.

Robert Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society: Series B (Methodological)*, 58(1):267–288, 1996. doi: 10.1111/j.2517-6161.1996.tb02080.x.

Yusuke Tsuzuku, Issei Sato, and Masashi Sugiyama. Lipschitz-margin training: Scalable certification of perturbation invariance for deep neural networks. In *Advances in Neural Information Processing Systems*, volume 31, 2018.

Kilian Q. Weinberger and Lawrence K. Saul. Distance metric learning for large margin nearest neighbor classification. *Journal of Machine Learning Research*, 10(9):207–244, 2009. URL http://jmlr.org/papers/v10/weinberger09a.html.

Kewei Xu, Xiaoben Lu, Shuofei Qiao, Zihan Ding, Haoming Xu, Lei Liang, and Ningyu Zhang. LongDS-Bench: On the failure of long-horizon agentic data analysis. *arXiv preprint arXiv:2605.30434*, 2026. URL https://arxiv.org/abs/2605.30434.

Manzil Zaheer, Satwik Kottur, Siamak Ravanbakhsh, Barnabas Poczos, Ruslan Salakhutdinov, and Alexander J. Smola. Deep sets. In *Advances in Neural Information Processing Systems*, volume 30, 2017.

Dan Zhang, Madelon Hulsebos, Yoshihiko Suhara, Çağatay Demiralp, Jinfeng Li, and Wang-Chiew Tan. Sato: contextual semantic type detection in tables. *Proc. VLDB Endow.*, 13(12):1835–1848, July 2020. ISSN 2150-8097. doi: 10.14778/3407790.3407793. URL https://doi.org/10.14778/3407790.3407793.

A Additional Graphs

A.1 Per-Dataset Performance

Figure 3 disaggregates the aggregate DAB results in Section 4 by dataset. Each cell reports dataset-level Pass@1 for the corresponding system configuration, making visible where aggregate differences arise rather than treating the benchmark as a single homogeneous task family.

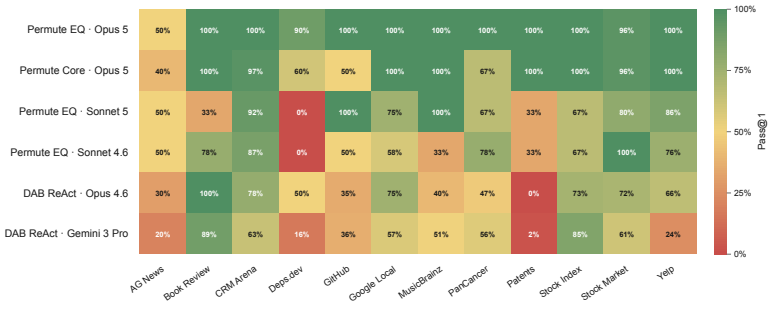

Figure 3: **Per-dataset DAB Pass@1 across evaluated configurations.** Each cell reports the Pass@1 percentage for one of the 12 benchmark datasets. The full Opus 5 realization attains perfect Pass@1 on nine datasets and exceeds the corresponding Core configuration on AG News, CRM Arena, Deps.dev, GitHub, and PanCancer, while matching it on the remaining seven datasets.

The largest gaps relative to Core occur on GitHub ($50\% \rightarrow 100\%$), PanCancer ($67\% \rightarrow 100\%$), and Deps.dev ($60\% \rightarrow 90\%$). These differences are descriptive; the aggregate evaluation remains the primary comparison reported in the main text.

A.2 Cost, resources, and consistency

The following diagnostics characterize the operating point used in our benchmark runs rather than a token-optimized deployment. Permute EQ was configured for conservative finalization: after locating a candidate answer, the system could continue to retrieve corroborating evidence, challenge competing interpretations, recompute intermediate results, and pass the answer through explicit arbitration and criticism before returning it. In several manually inspected traces, the eventual correct answer appeared within approximately 3–5 tool calls, after which the system used on the order of 15 additional calls to verify the result. This trace observation is qualitative rather than a benchmark-wide average, but it illustrates that a substantial portion of the observed runtime can arise after answer discovery.

We retained these guardrails deliberately. Our objective in this evaluation was to measure the complete system under an accuracy- and safety-oriented verification policy, not to minimize token use through aggressive early stopping. We report the resulting resource consumption because it exposes the cost of that choice and identifies where further optimization is possible. The current measurements should therefore be read as a high-reliability operating point rather than as a claim of token optimality.

Quality and monetary cost. Figure 4 compares dataset-macro quality with the mean cost of one complete trial. The horizontal axis shows mean cost per complete trial, and movement toward the upper-left corresponds to greater capability at lower cost. Across the EQ configurations, the figure shows the practical effect of downstream reasoner choice. Sonnet 5 reaches 82.9% stratified Pass@1 at a mean cost of $\$41.10$, nearly matching the 84.1% Core result while remaining below Core's reported mean cost of approximately $\$56$. Permute EQ with Opus 5 reaches the highest measured quality, 94.7%, at a higher mean cost of $\$82.82$.

Only the Core-versus-EQ Opus 5 comparison holds the downstream reasoner and agentic setting fixed. Comparisons among the Opus and Sonnet configurations instead illustrate different quality–cost operating points. The reported points have not been jointly optimized over reasoner choice, verification depth, stopping policy, or token budget, and Core's cost remains approximate because its accepted artifacts omit the corresponding cost fields.

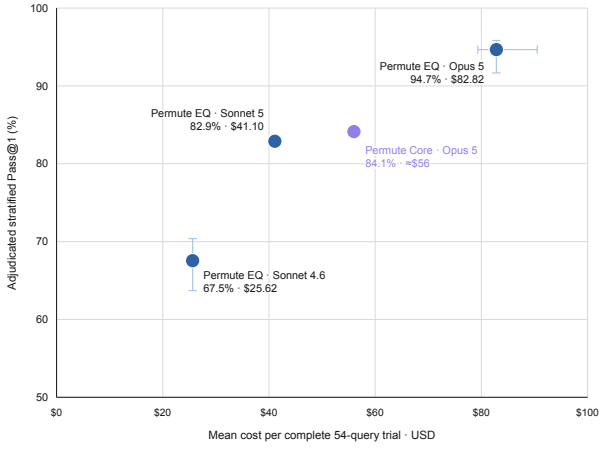

Figure 4: Dataset-macro DAB quality versus mean cost per complete 54-query trial for the evaluated configurations. Permute EQ points use adjudicated quality and measured scorecard cost. Core's accepted artifacts omit cost fields, so its point uses the reported mean of approximately $\$56$ per run for the final max-reasoning configuration. The displayed points were not optimized jointly for token efficiency and verification depth.

Cumulative token and tool usage. Figure 5 reports mean cumulative token and tool-call usage over a complete 54-query trial. These totals include not only the calls used to locate and compute an answer, but also subsequent corroboration, contradiction checks, recomputation, arbitration, and critic validation. They therefore measure the full conservative execution policy used in the reported runs.

This distinction is also important for interpreting the bounded evidence view. The bound applies to the evidence materialized for an individual reasoning step; it does not bound the cumulative number of scoped retrieval, execution, or verification steps performed across an entire trial. A system may therefore maintain bounded prompt-facing evidence while still accumulating substantial token and tool usage through repeated verification. The larger EQ totals should be interpreted in this context rather than as evidence that query conditioning failed to constrain the evidence exposed at any one step.

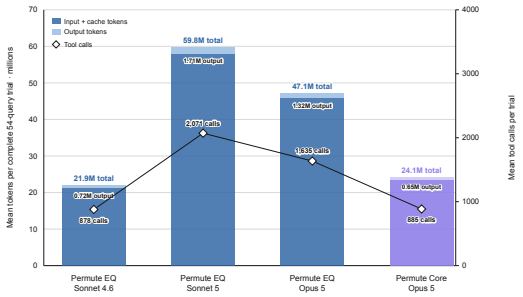

Figure 5: Mean token and tool-call usage per complete 54-query trial under the evaluated accuracy-first configurations. Each bar separates input and cache tokens from output tokens, while the line reports mean tool calls. The totals include conservative post-answer verification. Core's five reported usage summaries average 23.455 million input and cache tokens and 0.655 million output tokens, or 24.110 million total tokens.

Several of the current guardrails could in principle be reduced through confidence-dependent stopping, adaptive verification depth, fewer repeated challenge paths, or earlier termination once independent checks agree. Those alternatives were not evaluated here. The present results therefore identify an unoptimized point on a broader tradeoff among offline training duration, downstream reasoner capability, online token and tool use, monetary cost, and final-answer reliability. We hypothesize that more careful tuning of this frontier could retain comparable benchmark performance while materially reducing online cost, but establishing such a Pareto-efficient configuration requires a dedicated ablation.

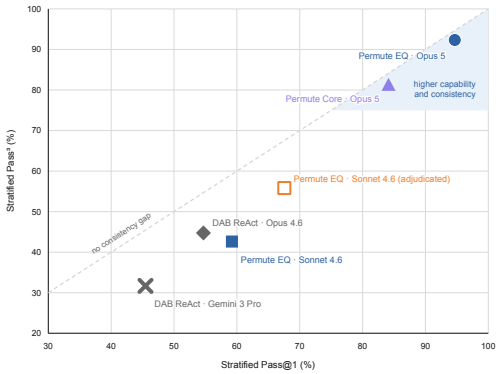

Figure 6: Dataset-macro capability versus repeated-trial consistency. Pass^3 requires a query to pass in all three sampled trials and is distinct from Pass@3. The diagonal denotes no consistency gap; points closer to the upper-right combine stronger average performance with greater trial-to-trial stability. For the 50-run Gemini result, the displayed consistency score averages over every three-run subset.

Capability and repeated-trial consistency. Figure 6 compares ordinary stratified Pass@1 with Pass^3. Pass^3 is not Pass@3: a query contributes a success only when it passes in all three sampled trials, after which the query-level indicators are aggregated using the same dataset-macro structure as Pass@1. The horizontal axis therefore measures average capability, while the vertical axis measures how much of that capability is retained consistently across repeated trials.

Since a query that passes all three trials must also pass an individual trial, Pass^3 cannot exceed Pass@1. The diagonal represents no consistency gap, while vertical distance below it reflects run-to-run fragility. Permute EQ with Opus 5 occupies the upper-right region and lies close to the diagonal, combining the highest average capability with strong three-run consistency. This behavior is consistent with the system's conservative finalization policy, although the present experiments do not isolate how much of the consistency is attributable to verification depth, the learned equivalence representation, or their interaction. A verification-depth ablation would be required to make that causal attribution.

B Assignment Structure, Routing, and Stability

We separate elementary properties of the assignment maps from the routing factorization and conditional stability guarantees. Unless a change of realization is explicitly considered, the trained parameters and the prototype bank for D are fixed. All weights and model outputs below are finite. The statements characterize the disclosed mathematical formulation; they do not assume access to the mechanisms constructing $\omega_D^{\pm}$.

B.1 Assignment Fibers and Soft Membership

Equivalence and quotient. Because $\mathcal{C}$ is finite and nonempty, the maximizer set of every $\rho_D(e)$ is nonempty. The fixed selector T makes κ_D single-valued. Reflexivity, symmetry, and transitivity of equality therefore imply the same three properties for $e \sim_D e' \iff \kappa_D(e) = \kappa_D(e')$. For every $e \in \mathcal{E}(D)$,

$$[e]_D = \{e' : \kappa_D(e') = \kappa_D(e)\} = Q_{\kappa_D(e),D}.$$

Every evidence object belongs to its own assignment fiber, and two fibers intersect only if their identity labels agree. The nonempty fibers consequently partition $\mathcal{E}(D)$. The map $[e]_D \mapsto \kappa_D(e)$ is well-defined, injective, and surjective onto $\kappa_D(\mathcal{E}(D))$, proving the stated quotient correspondence. These facts hold for any single-valued assignment map; no novelty is claimed for this construction.

Prototype decision cells. Let $z_D(e) \stackrel{\text{def}}{=} z_e$ name the embedding component of $\psi_\theta(\Phi_D(e))$. The prototype construction is well-defined: $s_{c,D}^{\sup} \geq 0$, its positive branch has a nonzero denominator,

and its zero-support branch uses the prior. The prior and support precision matrices are positive definite, as are their convex combinations. Consequently every $d_{c,D}(z)$ is finite and nonnegative. Since $\tau > 0$, maximizing the distance softmax is equivalent to minimizing $d_{c,D}$, including all ties. Hence

$$\kappa_D = h_D \circ z_D, \qquad V_{c,D} \stackrel{\text{def}}{=} h_D^{-1}(\{c\}), \qquad Q_{c,D} = z_D^{-1}(V_{c,D}).$$

The final equality follows because $\kappa_D(e) = c \iff h_D(z_D(e)) = c \iff z_D(e) \in V_{c,D}$. Thus $P_{c,D}$ is a prototype realization, $V_{c,D}$ is a decision cell obtained by comparing all prototypes, and $Q_{c,D}$ is its evidence-space preimage. Neither the identity label nor an individual prototype is the set-valued equivalence class.

Proposition 2 (Membership pseudometric)**.** *The function $d_D^{\mathrm{mem}}(e, e') = \|\rho_D(e) - \rho_D(e')\|_2$ is a pseudometric. Its zero-distance equivalence classes refine the hard-assignment fibers, and the refinement can be strict.*

Proof. Nonnegativity, zero self-distance, and symmetry follow from the Euclidean norm. Its triangle inequality gives

$$\|\rho_D(e) - \rho_D(e'')\|_2 \le \|\rho_D(e) - \rho_D(e')\|_2 + \|\rho_D(e') - \rho_D(e'')\|_2.$$

Zero distance is equivalent to equality of the membership profiles, which implies equal hard readouts under the same T. Conversely, in a two-identity example, $(3/4, 1/4)$ and $(2/3, 1/3)$ have the same unique maximizer but differ by Euclidean distance $\sqrt{2}/12$. Thus hard equivalence need not imply zero membership distance. This also shows why the membership pseudometric need not descend to a metric on the coarser hard quotient. □

Remaining well-definedness checks. The finite softmax normalizers defining ρ_D, β, and γ are strictly positive, so the resulting vectors belong to their respective simplices. Each summand of $\mathcal{L}_+$ and $\mathcal{L}_-$ is nonnegative. An opposing-evidence term vanishes when its squared distance reaches ξ_-; it is strictly positive below that margin only when its weight $\omega_D^-(e, c)$ is positive. None of these observations guarantees that optimization attains every desired separation.

B.2 Identity-Factorized Routing

Fix the shared identity inventory, query projector g_ϕ, query metric, query prototypes, and compatibility matrix M. As in Section 3.4, these query-side objects receive no additional dataset-dependent input. For fixed D, B, take conditioning to be deterministic, or condition on a fixed realization of its randomness.

Proposition 3 (Identity-factorized evidence selection)**.** *With $\Gamma_M(b) = \mathrm{softmax}(M^\top b)$ and $\mathsf{F}_{D,B}(\pi) = \mathsf{C}_B(\mathsf{S}_D, \pi)$,*

$$\mathcal{V}_B(D, q) = \mathsf{F}_{D,B}\big(\Gamma_M(\beta(q))\big).$$

For any $q, q' \in \mathcal{Q}$,

$$\beta(q) = \beta(q') \implies \gamma(q) = \gamma(q') \implies \mathcal{V}_B(D, q) = \mathcal{V}_B(D, q').$$

For the same query and fixed query-side model, replacing D by another environment D' over the same identity inventory leaves $\gamma(q)$ unchanged; only the dataset-specific materialization map is replaced.

Proof. Substitution of $\gamma = \Gamma_M \circ \beta$ into the definition of $\mathcal{V}_B$ gives the factorization. Equal β profiles have equal images under Γ_M; equal activation profiles then have equal images under the fixed map $\mathsf{F}_{D,B}$. Neither $\beta(q)$ nor M depends on D under the stated assumptions, so changing S_D affects $\mathsf{F}_{D,B}$ but not $\gamma(q)$. □

This is an architectural separation property, not a statistical sufficiency claim for the answer. The requested evidence may be absent in D', the materialized views may differ, and the Arbiter and agent still receive the original query. If the query encoder, compatibility map, identity inventory, or query-side inputs change with the dataset, the dataset-invariance conclusion does not follow.

Why the query-side hard class is not enough. A deterministic hard readout of β could itself define query classes, but routing uses the full profile. For example, with $\mathcal{K} = \mathcal{C} = \{1, 2\}$ and $M = I_2$, the profiles $b = (3/4, 1/4)^\top$ and $b' = (2/3, 1/3)^\top$ share their hard maximizer, while $\Gamma_M(b) \neq \Gamma_M(b')$: their output odds are respectively $\exp(1/2)$ and $\exp(1/3)$. Thus a routing map on hard query classes would require an additional constancy condition that the present framework does not impose. The compatibility matrix connects two soft representations; it does not declare query requirements equivalent to evidence identities.

B.3 Local Stability of Prototype Assignments

The proof applies the standard margin–Lipschitz certification principle (Tsuzuku et al., 2018) to the distances in Section 3.3. The square root $\Lambda_{c,D}^{1/2}$ is the unique symmetric positive-definite square root; $\|A\|_{\mathrm{op}} = \sup_{\|x\|_2=1} \|Ax\|_2$ is the Euclidean operator norm. In particular, $L_{c,D} = \sqrt{\lambda_{\max}(\Lambda_{c,D})} > 0$.

Proof of Proposition 1. Write $A_{c,D} = \Lambda_{c,D}^{1/2}$, so that $d_{c,D}(z) = \|A_{c,D}(z - \mu_{c,D})\|_2$. For any $z, z' \in \mathbb{R}^p$, the reverse triangle inequality gives

$$\begin{aligned} |d_{c,D}(z') - d_{c,D}(z)| &\leq \|A_{c,D}(z' - z)\|_2 \\ &\leq L_{c,D}\|z' - z\|_2. \end{aligned}$$

Fix e and its unique winning identity c_*. For each competitor $c \neq c_*$, put $\Delta_{c,D}(e) = d_{c,D}(z_e) - d_{c_*,D}(z_e) > 0$. Applying the preceding inequality to the competitor and winner separately,

$$\begin{aligned} d_{c,D}(z_e + \delta) - d_{c_*,D}(z_e + \delta) &\geq d_{c,D}(z_e) - L_{c,D}\|\delta\|_2 \\ &\quad - d_{c_*,D}(z_e) - L_{c_*,D}\|\delta\|_2 \\ &= \Delta_{c,D}(e) - (L_{c,D} + L_{c_*,D})\|\delta\|_2. \end{aligned}$$

The finite minimum defining $r_D(e)$ is positive. If $\|\delta\|_2 < r_D(e)$, the final expression is strictly positive for *every* competitor. Hence c_* remains the unique minimizer and $h_D(z_e + \delta) = c_*$. Taking $\delta = z_{e'} - z_e$ gives the statement about membership in $Q_{c_*,D}$. □

A simpler conservative radius. Define the nearest-competitor margin and maximum distance sensitivity by

$$m_D(e) \stackrel{\text{def}}{=} \min_{c \neq c_*} \Delta_{c,D}(e), \qquad L_D^{\max} \stackrel{\text{def}}{=} \max_{c \in \mathcal{C}} L_{c,D}.$$

Then $r_D(e) \geq m_D(e)/(2L_D^{\max})$ because every numerator is at least $m_D(e)$ and every denominator is at most $2L_D^{\max}$. A runner-up-specific sensitivity bound is not sufficient unless the other competitors are also controlled. At a tie this argument provides no positive radius. If $|\mathcal{C}| = 1$, the hard assignment is constant everywhere.

What is certified. The result preserves the selected identity, not its semantic correctness, and it does not assert equality of the full soft-membership vectors. It applies to an embedding perturbation with the entire bank fixed, including its means and precision matrices. If the embedding component $f_\theta : \mathbb{R}^m \to \mathbb{R}^p$ of ψ_θ is K_θ-Lipschitz on a feature neighborhood, with $K_\theta > 0$, the same argument certifies feature perturbations that remain in that neighborhood and have norm below $r_D(e)/K_\theta$. This is conditional on that Lipschitz bound and does not establish invariance to arbitrary raw-schema transformations.

B.4 From Signed-Evidence Losses to a Stability Margin

The current objective already supplies a conditional connection between evidence quality and local stability. This connection uses its existing losses, not a new loss or an additional training procedure. For a fixed evidence object e, define its contributions to those losses by

$$\ell_+(e) \stackrel{\text{def}}{=} \sum_{c \in \mathcal{C}} \omega_D^+(e, c) d_{c,D}^2(z_e),$$

$$\ell_-(e) \stackrel{\text{def}}{=} \sum_{c \in \mathcal{C}} \omega_D^-(e, c) [\xi_- - d_{c,D}^2(z_e)]_+^2.$$

Proposition 4 (Conditional loss-to-margin bound)**.** *Suppose* $|\mathcal{C}| \geq 2$ *and choose a candidate identity* c_0 *with* $w_+ \overset{def}{=} \omega_D^+(e, c_0) > 0$ *and* $w_- \overset{def}{=} \min_{c \neq c_0} \omega_D^-(e, c) > 0$*. Set*

$$a_e \overset{def}{=} \sqrt{\ell_+(e)/w_+}, \qquad b_e \overset{def}{=} \sqrt{\left[\xi_- - \sqrt{\ell_-(e)/w_-}\right]_+}.$$

If $b_e > a_e$*, then* c_0 *is the unique nearest identity and*

$$r_D(e) \geq \frac{b_e - a_e}{2L_D^{\max}} > 0.$$

Proof. All loss summands are nonnegative. The summand for c_0 in ℓ_+ gives $w_+ d_{c_0,D}^2(z_e) \leq \ell_+(e)$, hence $d_{c_0,D}(z_e) \leq a_e$. For each competitor $c \neq c_0$, its opposing weight is at least w_-, so

$$w_-[\xi_- - d_{c,D}^2(z_e)]_+^2 \leq \ell_-(e).$$

Taking square roots gives $[\xi_- - d_{c,D}^2(z_e)]_+ \leq \sqrt{\ell_-(e)/w_-}$. Since $x \leq [x]_+$, rearrangement yields

$$d_{c,D}^2(z_e) \geq \xi_- - \sqrt{\ell_-(e)/w_-}.$$

Combining this with $d_{c,D}^2(z_e) \geq 0$ gives $d_{c,D}(z_e) \geq b_e$. Thus every competitor exceeds the candidate distance by at least $b_e - a_e > 0$. The conservative radius bound above now proves the result. □

Supporting loss controls proximity to the candidate, opposing loss controls separation from covered competitors, and the precision norms control sensitivity to displacement. Positive opposing coverage of every competitor is essential for this particular uniform guarantee; a zero opposing weight provides no such control. Training need not attain $b_e > a_e$, and the bound does not establish that the evidence endorses the semantically correct identity.

B.5 When the Prototype Realization Also Changes

A dataset modification can change both the embedding and its prototype bank. The fixed-bank certificate must not be used unchanged in that case. For completeness, consider two banks in the same $\mathbb{R}^p$, with the same identity labels and positive-definite precisions. Write $A_c = \Lambda_{c,D}^{1/2}$ and $\widetilde{A}_c = \widetilde{\Lambda}_c^{1/2}$, and let $\widetilde{z}$ be the modified embedding. Suppose

$$\|\widetilde{z} - z_e\|_2 \leq \epsilon_z, \quad \|\widetilde{\mu}_c - \mu_{c,D}\|_2 \leq \epsilon_{\mu,c}, \quad \|\widetilde{A}_c - A_c\|_{\mathrm{op}} \leq \epsilon_{A,c}.$$

Define the nonnegative distance-drift bound

$$\eta_c \overset{\text{def}}{=} (L_{c,D} + \epsilon_{A,c})(\epsilon_z + \epsilon_{\mu,c}) + \epsilon_{A,c}\|z_e - \mu_{c,D}\|_2.$$

Indeed, the identity

$$\begin{aligned}\widetilde{A}_c(\widetilde{z} - \widetilde{\mu}_c) - A_c(z_e - \mu_{c,D}) = \widetilde{A}_c\big[(\widetilde{z} - z_e) - (\widetilde{\mu}_c - \mu_{c,D})\big] \\ + (\widetilde{A}_c - A_c)(z_e - \mu_{c,D})\end{aligned}$$

and the reverse triangle inequality imply $|\widetilde{d}_c(\widetilde{z}) - d_{c,D}(z_e)| \leq \eta_c$, where $\widetilde{d}_c(z) = \|\widetilde{A}_c(z - \widetilde{\mu}_c)\|_2$. Therefore the original unique winner c_* is preserved whenever

$$\Delta_{c,D}(e) > \eta_c + \eta_{c_*} \qquad \text{for every } c \neq c_*.$$

This extension identifies the additional quantities that must be controlled when support is recomputed. It does not assume that arbitrary changes of dataset automatically satisfy these bounds.

C Sample Execution Trace

Query. For LGG patients, compute the average log10-transformed expression of the IGF2 gene across different histology types. Include only valid IGF2 values and histology annotations not enclosed in square brackets. Report at least four decimal places.	
Permute EQ (correct) **Summary of Prompt:** Answer the query using only the provided evidence payload and DAB data. Use the available tools to verify the evidence and compute the final answer. *13 agent steps; 31 tool calls*	**Permute Core (incorrect)** **Summary of Prompt:** Answer the query without internet access using only the provided DAB files. Use the available tools to inspect the data and compute the final answer. *20 agent steps; 20 tool calls*
1. Evidence setup `[DuckDB ×2]` Read query interpretation and diagnostics. **2. Evidence setup** `[DuckDB ×2]` Read selected tasks, anchors, and resolved fields. **3. Exploration** `[DuckDB ×2]` Inspect clinical and RNA-seq schemas. **4. Exploration** `[DuckDB ×2]` Sample patient identifiers and expression records. **5. Exploration** `[DuckDB ×2]` Identify the LGG cohort and verify its text pattern. **6. Disambiguation** `[DuckDB ×2]` Count LGG patients and inspect ICD-O-3 groups. **7. Disambiguation** `[DuckDB ×2]` Compare histology fields and bracket patterns. **8. Exploration** `[DuckDB ×2]` Verify the IGF2 symbol and expression validity. **9. Execution** `[DuckDB ×2]` Join LGG patients to IGF2 and inspect sample types. **10. Execution** `[DuckDB]` Compute candidate aggregates at both histology grains. **11. Verification** `[DuckDB ×2]` Check joined records through alternate extracts. **12. Verification** `[calculator ×5]` Evaluate the unshifted log transform by code. **13. Execution** `[calculator ×5]` Compute the selected $\log_{10}(x+1)$ means by code. **Answer.** `9382/3` 2.7136; `9400/3` 2.6014; `9401/3` 2.5584; `9450/3` 2.6967; `9451/3` 2.5826.	**1. Exploration** `[semantic probe]` Profile clinical categories, including histology fields. **2. Exploration** `[semantic probe]` Profile RNA-seq sample types. **3. Disambiguation** `[semantic probe]` Compare the three candidate histology fields. **4. Exploration** `[semantic probe]` Search patient descriptions for cohort terms. **5. Exploration** `[DuckDB]` Inspect the clinical schema. **6. Exploration** `[DuckDB]` Sample RNA-seq expression records. **7. Exploration** `[DuckDB]` Sample clinical records and histology values. **8. Exploration** `[DuckDB]` Inspect diagnosis frequencies. **9. Exploration** `[DuckDB]` Test cohort phrase extraction. **10. Exploration** `[DuckDB]` Count glioma and lower-grade glioma records. **11. Exploration** `[DuckDB]` Enumerate glioma phrases in patient descriptions. **12. Execution** `[DuckDB]` Isolate LGG patients and group descriptive histology labels. **13. Exploration** `[DuckDB]` Inspect IGF2 coverage and sample types. **14. Execution** `[DuckDB]` Join LGG patients to valid IGF2 records. **15. Disambiguation** `[DuckDB]` Compute candidates for all three histology fields. **16. Verification** `[DuckDB]` Recompute descriptive-label means with an alternate join. **17. Verification** `[Python]` Independently reproduce the descriptive-label result. **18. Disambiguation** `[semantic probe]` Reinspect the candidate histology fields. **19. Execution** `[DuckDB]` Finalize the three-label aggregation.

Continued on next page

PANCANCER_ATLAS Q1 agent trajectories—continued	
Permute EQ (correct)	**Permute Core (incorrect)**
	20. Selection `[result gate]` Select the internally consistent result at the wrong grain. **Answer.** Oligoastrocytoma 2.7136; Oligodendroglioma 2.6825; Astrocytoma 2.5713. The required five ICD-O-3 groups are not reported.

Figure 7. Observed agent trajectories for PANCANCER_ATLAS Q1. Each numbered item represents one agent step; multipliers denote tool calls issued in parallel within that step. Both systems correctly identify the LGG cohort, join the clinical and expression data, and compute the requested transformation. They diverge at the grouping-field decision: Permute EQ selects the required ICD-O-3 histology groups, whereas Permute Core aggregates the same underlying records into three broader descriptive histology labels. Its numerical computation is therefore consistent, but its output is incorrect because it is reported at the wrong histology grain and omits the five required ICD-O-3 groups. The summaries describe visible actions and results, not hidden chain-of-thought. This selected example does not represent aggregate performance.

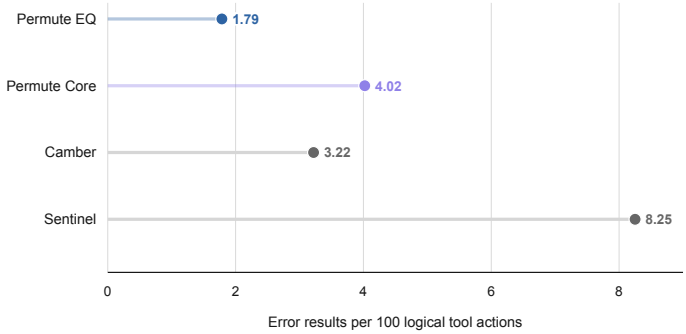

Figure 8: Aggregated tool errors taken from the tool traces