READER: DYNAMIC LLM PROVENANCE FROM QUERY-VARYING INTERACTIONS

Anonymous authors
Paper under double-blind review

ABSTRACT

Auditing the provenance of large language models (LLMs) is essential for verifying service claims and detecting model substitution. Yet API services rarely expose the internals required by white-box methods, while many black-box methods establish comparability through shared diagnostic prompts. Auditing query-varying interactions therefore requires comparable source evidence as observations accumulate. We formalize this setting as *dynamic black-box LLM provenance* within a fixed set of enrolled models. To make these interactions comparable, we introduce READER, which uses a frozen proxy LLM as a common measurement space and encodes the mean and coarse evolution of activations along each response into spectral fingerprints. A linear probe trained once at enrollment extracts source evidence that Bayesian accumulation combines across observations. On Agent500, our benchmark containing 50,000 responses from 100 models to 500 heterogeneous agent prompts, READER reaches 50.4% attribution accuracy from one response and 96.2% from 100 on unseen prompts, exceeding fine-tuned DeBERTa and LLM-DNA with frozen sentence encoders at the respective budgets. Experiments across proxy families further demonstrate strong cumulative attribution, while static analyses reveal model relationships in the same spectral representation. READER thus turns ordinary interactions into cumulative evidence for provenance auditing.

1 INTRODUCTION

Large language models increasingly operate as invisible infrastructure behind agents, application backends, and third-party API services (Yang et al., 2026). In this setting, independent evidence is needed to verify which model served a request. An intermediary may wrap a cheaper or unauthorized backend. A provider update may silently change capability or safety behavior. A forensic investigation may need to associate harmful output with its actual source. Verifying model identity supports reliable deployment and accountability as API behavior changes over time (Tschisgale & Wulff, 2026). An independent auditor usually lacks parameters, logits, gradients, decoding traces, and planted watermarks. The available evidence is the interaction itself: a prompt and the text returned by an unknown backend.

Intrinsic black-box methods establish identity or lineage through shared, crafted, or otherwise predefined prompt panels (Gubri et al., 2024; Pasquini et al., 2025; Nikolic et al., 2025; Yax et al., 2025; Wu et al., 2026). Repeated auditing with these panels incurs additional latency and API charges. Meanwhile, ordinary users and agents continuously produce records of query-varying interactions. We formalize *dynamic black-box LLM provenance*: after enrolling a fixed candidate ecosystem, attribute observed interactions as prompts vary and evidence accumulates. At any available query budget, attribution proceeds directly from traffic already produced in use. Figure 1 distinguishes this setting from white-box access and static black-box comparison.

An audit record may contain one interaction or hundreds, each addressing a different task. The representation must preserve source evidence across responses to different prompts, and the decision rule must accumulate this evidence as the record grows. Our key hypothesis is that a frozen proxy LLM supplies a common measurement space. When the proxy reads an interaction, the trajectory of its response states can reveal structure associated with source identity.

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.

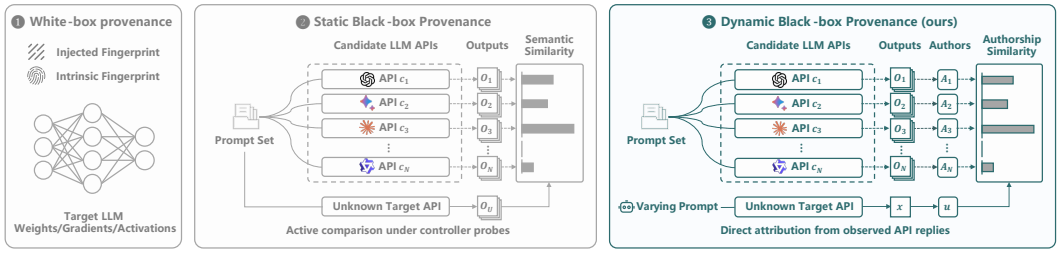


Figure 1: **Provenance settings.** White-box methods use target internals. Static black-box methods actively query candidate APIs with a shared panel. Dynamic provenance attributes observed, query-varying interactions after enrollment.

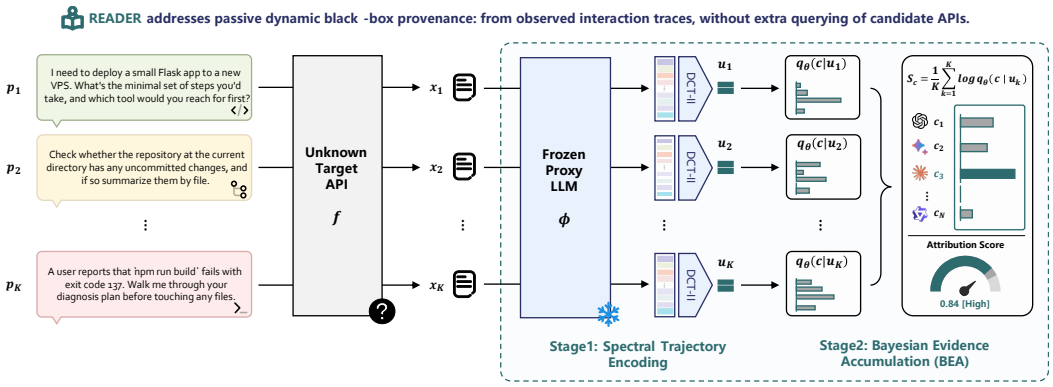


Figure 2: **READER pipeline.** A frozen proxy maps each interaction to a DC–AC trajectory fingerprint. One probe trained during enrollment reads source evidence from each response and accumulates it over any query budget.

We instantiate this idea with **READER**, short for *Response Evidence for Authorship Decoding via Extracted Representations* (Figure 2). READER retains the frozen proxy states aligned with response tokens and applies a length-normalized DCT. The DC coordinate captures mean activation across the response. The first AC coordinate captures its coarsest variation along the response. Together they form a trajectory fingerprint for source identity and model relationships. A linear probe converts one fingerprint into source evidence. Bayesian Evidence Accumulation then combines log posteriors across observations, allowing one probe trained during enrollment to serve every inference budget.

We evaluate this setting with **Agent500**, which crosses 500 agent prompts with 100 candidate sources, split evenly between local and API models, to produce 50,000 interactions. READER reaches 50.4% 100-way accuracy from one response and 96.2% from 100. Fine-tuned DeBERTa reaches 33.0% at the first budget, and LLM-DNA with frozen sentence encoders reaches up to 79.0% at the second. All four main proxy families reach at least 94.8% mean accuracy at $K = 100$. Controlled truncation and the 10,000 interactions in **Math100** test transfer across response lengths and query domains. Static relationship classification and neighborhood tests in the original space then examine the model relationships encoded by the spectral coordinates. We will release both benchmarks, target metadata, and the implementation.

Our contributions follow this progression. First, we formalize dynamic black-box provenance from query-varying interactions and growing evidence budgets. Second, we introduce a spectral proxy reader and a reusable evidence accumulator for individual responses. Third, Agent500, Math100, and the accompanying analyses establish 100-way attribution performance, evaluate transfer across response lengths and domains, and reveal model relationships in the shared measurement space.

2 Related Work

Provenance by design and internal access. Model identity can be established by planting a signal before deployment or by comparing model internals after release. Decoding-time watermarks impose detectable token-level statistics (Kirchenbauer et al., 2023; Kuditipudi et al., 2024). Training-time backdoors in embedding services and instruction-time fingerprints in generative LLMs create triggerable ownership signatures (Peng et al., 2023; Xu et al., 2024). Subsequent methods improve semantic conditioning, scalability, and robustness to model merging (Gloaguen et al., 2026; Nasery et al., 2025; Yamabe et al., 2025). A complementary white-box line derives fingerprints from parameters, gradients, or internal representations (Zeng et al., 2024; Wu et al., 2025). REEF, for example, compares suspect and reference activations using centered kernel alignment (Kornblith et al., 2019; Zhang et al., 2025). These methods establish provenance through intervention before deployment or access to the target model's internals. Dynamic black-box auditing begins after deployment, when the interaction record is the available evidence.

Black-box behavioral provenance. Intrinsic black-box methods infer identity or lineage from a model's observable behavior. The style of generated text and logit spaces exposed by APIs can reveal model fingerprints (McGovern et al., 2025; Yang & Wu, 2026). TRAP and LLMmap construct targeted or carefully crafted probes that amplify behavioral differences between candidate models (Gubri et al., 2024; Pasquini et al., 2025). Model Provenance Testing and Model Provenance Set formulate statistical decisions over candidate parents or sources (Nikolic et al., 2025; Qiu et al., 2026). PhyloLM and LLM DNA build functional representations for model similarity and evolutionary relationships (Yax et al., 2025; Wu et al., 2026). This literature establishes the value of controlled queries for revealing provenance structure. Shared prompt panels make model outputs comparable. Dynamic provenance must recover that comparability from query-varying interactions already observed in use, then strengthen a decision as additional interactions arrive. READER addresses both requirements through a shared response fingerprint and evidence reused across query budgets.

Proxy representations as evidence. Frozen neural states make latent attributes accessible to linear probes, whose controls distinguish between decodability and causal use (Hewitt & Liang, 2019; Ravichander et al., 2021). The linear representation hypothesis characterizes how such structure emerges (Park et al., 2024; Jiang et al., 2024). Work on superposition, dictionary learning, and representation editing further identifies behaviorally meaningful features (Elhage et al., 2022; Bricken et al., 2023; Zou et al., 2023; Rimsky et al., 2024). These include linguistic style and emotion (Lai et al., 2024; Tak et al., 2025), with broader evidence reviewed by Zhang et al. (2026). READER applies this view across models: a frozen proxy reads an observed interaction and turns its response trajectory into an external measurement of source behavior. Spectral coordinates retain global location and coarse evolution, yielding comparable evidence for dynamic attribution.

3 Method

READER attributes black-box generations by using a frozen proxy language model to read evidence of source identity. Its two stages address the two requirements of dynamic provenance. *Spectral Trajectory Encoding* extracts a comparable fingerprint from responses to different queries. *Bayesian Evidence Accumulation* converts each fingerprint into source evidence and combines successive measurements under a changing query budget.

3.1 Dynamic Provenance Setting

Let $\mathcal{C} = \{1, \dots, C\}$ be a fixed ecosystem of candidate source models. An unknown source $y \in \mathcal{C}$ receives prompts $p_k \sim \mathcal{P}$ and generates responses $x_k \sim P_y(\cdot \mid p_k)$, where P_y is its conditional generation distribution. Prompts vary across interactions. Given a query budget K, dynamic provenance seeks

$$\widehat{y} = \arg\max_{c \in \mathcal{C}} p(c \mid \mathcal{X}_{1:K}), \qquad \mathcal{X}_{1:K} = \{(p_k, x_k)\}_{k=1}^{K}. \tag{1}$$

Enrollment provides labeled responses from the candidate ecosystem. Evaluation uses responses to unseen prompts. READER receives the observed prompt and generated text as its complete audit record.

For each observed pair (p, x), a frozen proxy model ϕ processes the complete prompt and response. At proxy layer ℓ, READER retains the hidden states aligned with the response tokens:

$$\mathbf{H}^{(\ell)}(p, x) = \left[\mathbf{h}_0^{(\ell)}, \ldots, \mathbf{h}_{T-1}^{(\ell)}\right]^\top \in \mathbb{R}^{T \times d}, \tag{2}$$

where T is the response length under the proxy tokenizer and d is the hidden dimension. The prompt supplies context to the proxy, and the trajectory encoder summarizes states at response positions.

3.2 Stage 1: Spectral Trajectory Encoding

Across query-varying interactions, evidence of source identity can persist in both the global location of a proxy trajectory and its coarse evolution across tokens. READER captures these signals by projecting all valid response states onto the first two modes of a length-normalized DCT-II. For $r \in \{0, 1\}$, it computes

$$\mathbf{z}_r(p, x) = \frac{\gamma_r}{T} \sum_{t=0}^{T-1} \mathbf{h}_t^{(\ell)} \cos\left[\frac{\pi r}{T}\left(t + \frac{1}{2}\right)\right], \tag{3}$$

with $\gamma_0 = 1$ and $\gamma_1 = \sqrt{2}$. Equation 3 is the orthonormal DCT-II coefficient divided by $\sqrt{T}$, preventing response length from mechanically scaling its magnitude.

The zeroth mode is the DC component,

$$\mathbf{z}_0(p, x) = \frac{1}{T} \sum_{t=0}^{T-1} \mathbf{h}_t^{(\ell)}, \tag{4}$$

and records mean activation across the response. The first AC component $\mathbf{z}_1$ measures the coarsest variation along the response. Its temporal weights sum to zero, making it invariant to a constant offset applied to the entire trajectory. For a single-token response, $\mathbf{z}_1 = \mathbf{0}$. Modes 0 and 1 therefore form the minimal spectral pair that retains the trajectory mean and its coarsest variation. Higher modes encode progressively finer variation across tokens and are studied separately in the ablation.

The response fingerprint concatenates the two coefficients:

$$\mathbf{u}(p, x) = \begin{bmatrix}\mathbf{z}_0(p, x) \\ \mathbf{z}_1(p, x)\end{bmatrix} \in \mathbb{R}^{2d}. \tag{5}$$

This unprojected DC–AC representation is deterministic, uses every response token, and leaves the proxy parameters fixed. By preserving both mean activation and coarse variation, the same DC–AC fingerprint supports dynamic source attribution and static relationship auditing through a separate readout for each task. Stage 2 develops the dynamic readout.

3.3 Stage 2: Bayesian Evidence Accumulation

Each fingerprint supplies a source measurement from one observed interaction. READER converts these measurements into evidence over candidate sources before combining them in decision space. Let $\mathcal{D}_{\text{tr}} = \{(\mathbf{u}_i, y_i)\}_{i=1}^N$ denote the enrollment fingerprints $\mathbf{u}_i = \mathbf{u}(p_i, x_i)$ and source labels $y_i \in \mathcal{C}$. A coordinate standardizer g_ψ, fitted on this split, normalizes each fingerprint before a multinomial linear probe:

$$q_\theta(c \mid \mathbf{u}) = \operatorname{softmax}\left(\mathbf{W}^\top g_\psi(\mathbf{u}) + \mathbf{b}\right)_c. \tag{6}$$

Here, $\mathbf{W} \in \mathbb{R}^{2d \times C}$, $\mathbf{b} \in \mathbb{R}^C$, and $\theta = (\mathbf{W}, \mathbf{b})$. Since $\mathbf{u}(p, x)$ includes prompt context, we use $q_\theta(c \mid \mathbf{u}(p, x))$ to approximate the source posterior conditioned on the prompt. The probe minimizes the regularized cross-entropy

$$\mathcal{L}(\theta) = -\frac{1}{|\mathcal{D}_{\text{tr}}|} \sum_{(\mathbf{u}_i, y_i) \in \mathcal{D}_{\text{tr}}} \log q_\theta(y_i \mid \mathbf{u}_i) + \frac{\lambda}{2}\|\mathbf{W}\|_F^2. \tag{7}$$

where $\|\cdot\|_F$ is the Frobenius norm and $\lambda \geq 0$ controls weight decay. Training on individual responses makes q_θ reusable across query budgets. Appendix B.1 specifies standardization and optimization.

For a query set $\mathcal{X}_{1:K}$, write $p_{1:K} = (p_1, \ldots, p_K)$, $x_{1:K} = (x_1, \ldots, x_K)$, $\mathbf{u}_k = \mathbf{u}(p_k, x_k)$, and $\mathbf{u}_{1:K} = (\mathbf{u}_1, \ldots, \mathbf{u}_K)$. We assume a stateless query protocol: conditioned on the source and the observed prompts, target requests do not share conversational or latent generation state. Therefore,

$$p(x_{1:K} \mid p_{1:K}, c) = \prod_{k=1}^{K} p(x_k \mid p_k, c), \quad \Longrightarrow \quad p(\mathbf{u}_{1:K} \mid p_{1:K}, c) = \prod_{k=1}^{K} p(\mathbf{u}_k \mid p_k, c). \tag{8}$$

Conditioning on the observed prompts accommodates query-varying interactions as long as prompt selection is source independent. Let $\pi_c > 0$ denote both the class prior represented in enrollment and the source prior used at inference. Our class-balanced setting gives $\pi_c = 1/C$. Substituting Eq. 6 for the source posterior conditioned on the prompt and applying the corresponding prior correction yields the composite log-posterior score (Appendix D.1)

$$S_c(\mathcal{X}_{1:K}) = \frac{1}{K} \sum_{k=1}^{K} \log(q_\theta(c \mid \mathbf{u}_k) + \varepsilon) - \frac{K-1}{K} \log \pi_c, \tag{9}$$

where $\varepsilon > 0$ provides numerical stability. READER predicts

$$\widehat{y} = \arg\max_{c \in \mathcal{C}} S_c(\mathcal{X}_{1:K}). \tag{10}$$

For a class-balanced ecosystem, the prior term is constant across sources, so Eq. 9 becomes mean log-posterior accumulation. At $K = 1$, this is the single-response probe. Larger budgets reuse the same probe and add evidence in log space.

The proxy layer ℓ is selected by prompt-grouped validation within enrollment and fixed before test evaluation (Appendix A.3). Within each split, enrollment data fit the standardizer, linear probe, and class priors. The spectral coefficients cost $O(Td)$ per response and produce a fixed $2d$-dimensional fingerprint. The dense probe costs $O(Cd)$ per response, followed by $O(KC)$ accumulation. The frozen proxy forward pass dominates inference cost in practice (Appendix J.2).

4 Experiments

4.1 Experimental Setup

Tasks. Agent500 is our primary dynamic attribution benchmark. It contains 500 agent-style prompts, each answered by the same 100 candidate source models (50 locally hosted and 50 accessed by API). Responses are generated under a common temperature of 0.7 and top-p of 0.95. Every method observes the prompt–response pair. Math100 and controlled response truncation test how the fixed Agent500 system transfers across domain and length shifts. From Bench-A, introduced with MPT (Nikolic et al., 2025), we construct a balanced relationship classification task over 116 model pairs from 67 unique models. A *parent group* comprises a designated parent checkpoint and its derivatives. Positive pairs connect a parent to a derivative, and negative pairs draw models from different groups. The task spans nine parent groups, with 600 prompt-aligned responses per model.

Proxies and baselines. The main READER results use four instruction-tuned proxies from distinct families: Llama-3.1-8B-Instruct, Gemma-4-12B-it, Ministral-3-8B-Instruct-2512, and Qwen3.5-9B. We select one layer per proxy on prompt-disjoint enrollment data and freeze it thereafter. Dynamic baselines include random guessing, a fine-tuned DeBERTa-v3-large text classifier, and dynamic adaptations of LLMmap (Pasquini et al., 2025) and LLM-DNA with three frozen sentence encoders (Wu et al., 2026). Their published protocols use a fixed shared prompt panel, so Agent500 evaluates adaptations that score query-varying interactions one response at a time. Static baselines are MPT, PhyloLM (Yax et al., 2025), and the original LLM-DNA concat–projection–centroid pipeline.

Protocol. Agent500 uses five-fold prompt-grouped cross-validation: all 100 responses to one prompt remain in the same fold. Within each fold, the standardizer and probe are fitted only on enrollment prompts. Query groups are sampled with three fixed seeds for each $K \in \{1, 5, 10, 20, 50, 100\}$. We report top-1 accuracy and macro-F1. Macro-F1 averages one-vs-rest F1

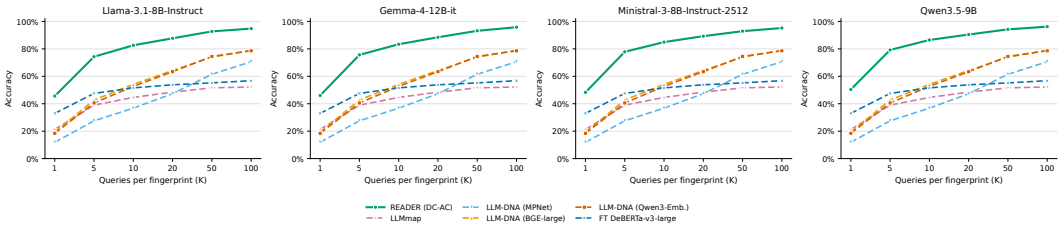


Figure 3: **Accuracy across query budgets on Agent500.** Each panel uses a distinct frozen proxy. One READER probe trained on individual responses is reused for every K. Curves are means under the prompt-grouped protocol, with LLMmap and LLM-DNA adapted from their fixed-panel settings.

over all 100 sources, giving equal weight to rare and difficult sources. The default static protocol, pair holdout, uses the same 20 fixed stratified 4:1 splits for every method and reports accuracy, precision, recall, F1, and AUC. Table captions specify the aggregation protocol. Appendices A–C detail benchmark construction, evaluation protocols, baseline adaptations, and target rosters.

4.2 Dynamic 100-Way Attribution

Table 1 gives the endpoint results for the two requirements of dynamic provenance: reading evidence from one previously unseen prompt and strengthening the decision as responses to different queries become available. Figure 3 resolves the full budget trajectory.

Table 1: **Agent500 100-way attribution.** Mean±standard deviation over five prompt-grouped folds; Random uses 20,000 Monte Carlo trials at fold-sized test budgets. M-F1 denotes macro-F1. †Dynamic adaptation of a fixed-panel method.

Method	Acc.@1	M-F1@1	Acc.@100	M-F1@100
Random	0.010±0.001	0.010±0.001	0.010±0.010	0.007±0.008
LLMmap†	0.212±0.003	0.189±0.005	0.522±0.040	0.423±0.039
DNA / MPNet†	0.119±0.004	0.109±0.005	0.706±0.043	0.644±0.045
DNA / BGE†	0.191±0.005	0.188±0.005	0.790±0.028	0.739±0.034
DNA / Qwen-Emb.†	0.185±0.006	0.192±0.004	0.786±0.012	0.735±0.016
FT DeBERTa	0.330±0.011	0.295±0.011	0.568±0.029	0.471±0.030
READER / Llama	0.456±0.011	0.455±0.011	0.948±0.013	0.931±0.018
READER / Gemma	0.460±0.012	0.457±0.012	0.958±0.015	0.945±0.019
READER / Ministral	0.483±0.011	0.481±0.011	0.952±0.017	0.939±0.021
READER / Qwen	**0.504±0.011**	**0.503±0.010**	**0.962±0.024**	**0.952±0.029**

From a single response, READER with Qwen reaches 50.4% accuracy, a gain of 17.4 points over fine-tuned DeBERTa and more than twice LLMmap. This result generalizes across readers, with all four READER variants exceeding 45%. Accumulation then resolves sources that remain ambiguous from one response. At $K = 100$, READER with Qwen reaches 96.2%, exceeding LLM-DNA with frozen sentence encoders by at least 17.2 points. Every proxy reaches at least 94.8% mean accuracy. Figure 3 traces the same advantage through every intermediate budget. A probe trained once therefore converts a growing interaction record into steadily stronger attribution. Appendices E.1–E.2 extend these results to all eight proxies and report computational profiles.

4.3 Transfer under Length and Domain Shifts

We next freeze each method's Agent500 estimator, including preprocessing and learned parameters, and vary only the evaluation responses. Table 2 compares response truncation and transfer to Math100 across the four proxy readers and baselines. Appendix E.5 extends the analysis across query budgets on Math100.

READER achieves the highest accuracy in all three conditions and the highest macro-F1 at 64 tokens and on Math100. At 64 tokens, all four proxies exceed fine-tuned DeBERTa on both metrics, with

Table 2: **Transfer without retraining at $K = 100$.** The first two column groups truncate Agent500 responses. All methods reuse their clean Agent500 estimators, and Math100 results average five fold models. Random reuses the same reference from 20,000 Monte Carlo trials across conditions. Bold and underlined values indicate the best and second best results in each column. †Dynamic adaptation of a fixed-panel method.

	32 tokens		64 tokens		Math100	
Method	**Acc.**	**M-F1**	**Acc.**	**M-F1**	**Acc.**	**M-F1**
Random	0.010	0.007	0.010	0.007	0.010	0.007
LLMmap†	0.058	0.046	0.144	0.109	0.128	0.065
DNA / MPNet†	0.034	0.029	0.026	0.018	0.046	0.018
DNA / BGE†	0.026	0.011	0.038	0.032	0.102	0.055
DNA / Qwen-Emb.†	0.014	0.006	0.020	0.012	0.142	0.077
FT DeBERTa	0.158	**0.143**	0.278	0.259	0.232	0.151
READER / Llama	**0.185**	0.142	0.407	0.343	0.262	0.170
READER / Gemma	0.170	0.132	0.429	0.379	0.204	0.127
READER / Ministral	0.138	0.095	0.368	0.307	0.282	0.191
READER / Qwen	0.159	0.111	**0.505**	**0.439**	**0.330**	**0.242**

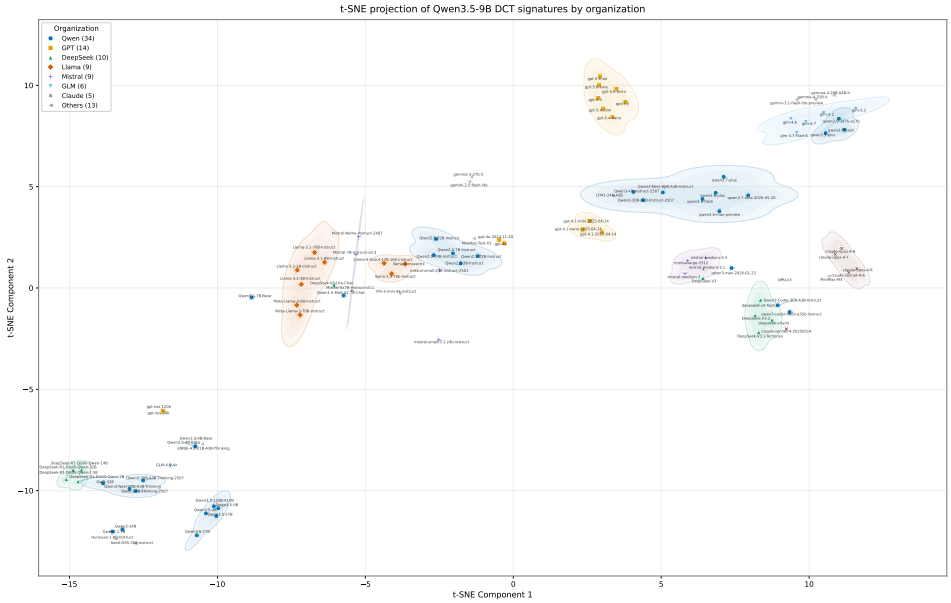


Figure 4: **Geometry of model signatures.** A cosine t-SNE of 100 model signatures read by Qwen3.5-9B, with complete model labels. Contours show dense regions within provider families. Local neighborhoods depict behavioral similarity on Agent500.

Qwen reaching 50.5% accuracy versus 27.8%. At 32 tokens, DeBERTa attains the highest macro-F1. On Math100, READER with Qwen reaches 33.0% accuracy versus DeBERTa's 23.2%, with greater variation across proxy readers. Relative to the clean results in Table 1, absolute accuracy declines under these shifts. READER's leading results across five of the six metrics support continued attribution with the reader fixed at enrollment, as response lengths and query domains change.

4.4 Static Relationship Auditing

Having evaluated source attribution, we next ask whether the same spectral coordinates preserve relationships among sources. We use the balanced task derived from Bench-A. For each pair, we compute prompt-aligned cosine similarity in the DC and first AC blocks, average both scores over prompts, and fit a linear SVM on the resulting two-dimensional feature. This static readout is trained

solely from the derived pair labels. Table 3 compares it with specialized relationship baselines under pair holdout. Appendix F.1 evaluates transfer under model holdout and parent group holdout. Appendix F.2 provides the complete model and pair roster. Section 4.5 then identifies which spectral block supports each task.

Table 3: **Static relationship classification under pair holdout.** READER uses a readout over the two DC–AC cosine scores. Values are mean±standard deviation over the same 20 stratified splits of 116 balanced model pairs from Bench-A.

Method	Acc.	Prec.	Rec.	F1	AUC
MPT	0.742±0.061	0.943±0.082	0.521±0.123	0.660±0.105	0.746±0.085
PhyloLM	0.758±0.057	**0.944±0.073**	0.554±0.113	0.690±0.093	0.793±0.078
DNA / MPNet	0.621±0.063	0.650±0.091	0.554±0.119	0.589±0.080	0.675±0.074
DNA / BGE	0.631±0.076	0.629±0.078	**0.646±0.139**	0.631±0.096	0.722±0.075
DNA / Qwen-Emb.	0.704±0.090	0.835±0.162	0.554±0.116	0.650±0.096	0.723±0.086
READER / Llama	**0.781±0.059**	0.928±0.079	0.617±0.127	**0.731±0.093**	**0.828±0.065**
READER / Gemma	0.694±0.068	0.730±0.091	0.629±0.125	0.668±0.086	0.768±0.078
READER / Ministral	0.754±0.063	0.906±0.089	0.575±0.131	0.693±0.099	0.811±0.063
READER / Qwen	0.767±0.062	0.907±0.088	0.600±0.122	0.714±0.096	0.809±0.074

Benchmark comparison. READER with Llama obtains the highest accuracy (78.1%), F1 (73.1%), and AUC (82.8%). PhyloLM retains the highest precision. Together, these results show that the spectral coordinates used for source attribution also support relationship classification through a linear readout.

4.5 Ablation Studies

To understand which design choices support these results, we isolate the spectral representation shared by both tasks and the evidence accumulation used for dynamic attribution.

Stage 1: Spectral Trajectory Encoding. We first vary how the proxy states are summarized while keeping the remaining pipeline fixed within each task (Table 4). DC–AC outperforms final token and coordinate maximum pooling across all reported metrics. Among the pooling concatenations with the same $2d$ dimension, mean + maximum achieves slightly higher accuracy from one response on Agent500, while DC–AC performs better at $K = 100$, on Math100, and on the pair task. Within this representation, removing DC reduces Agent500 accuracy by 9.3 points at $K = 1$. Removing first AC reduces pair accuracy by 20.7 points and Math100 accuracy by 3.8 points at $K = 100$. Retaining both coordinates therefore preserves complementary evidence for attribution and model relationships in a compact $2d$ measurement space. Appendix G.1 gives the full component analysis and matched protocol, and Appendix G.2 extends the temporal comparison on Agent500.

Stage 2: Bayesian Evidence Accumulation. We then freeze the DC–AC source heads and change only how their evidence is combined. Table 5 compares BEA with majority voting and probability averaging on identical query groups. All three agree at $K = 1$. BEA achieves higher mean accuracy than both alternatives at every tested budget above one. Its gain over probability averaging is 4.5 percentage points at $K = 5$ and 1.3 points at $K = 100$. With the source evidence held fixed, these gains isolate the benefit of BEA even when only a few interactions are available. Appendix G.3 provides the results for each proxy.

4.6 Geometry of Model Signatures

Beyond source attribution and pairwise classification, we examine how the shared spectral space organizes models by their response behavior. We average each source's 500 DC–AC fingerprints, normalize the two blocks, and compute cosine distance. Figure 4 visualizes these signatures with t-SNE. GPT-5 variants form a compact neighborhood, while Qwen models span several local regions, some overlapping with other providers.

Table 4: **Stage 1: Spectral Trajectory Encoding.** Means across four proxies. Agent500 uses five prompt folds, and Math100 evaluates the frozen fold models without retraining. Pair scores average twenty fixed splits. Every variant shares the proxy layer, response tokens, splits, and training recipe within each task. Bold and underlined values indicate the best and second best results in each column.

	Agent500		Math100		Pair task	
Representation	**Acc.@1**	**Acc.@100**	**Acc.@1**	**Acc.@100**	**Acc.**	**AUC**
Final token	0.159	0.659	0.067	0.158	0.645	0.731
Coordinate maximum	0.410	0.930	0.158	0.251	0.517	0.605
Mean + final	0.386	0.896	0.148	0.229	<u>0.649</u>	0.727
Mean + maximum	**0.478**	0.947	<u>0.176</u>	<u>0.261</u>	0.533	0.621
Final + maximum	0.355	0.886	0.142	0.242	0.636	0.719
DC–AC (full)	<u>0.476</u>	**0.956**	**0.182**	**0.271**	**0.748**	**0.804**
w/o DC (first AC only)	0.383	0.939	0.150	0.254	**0.748**	<u>0.800</u>
w/o first AC (mean / DC)	0.474	<u>0.955</u>	0.165	0.233	0.542	0.621

Table 5: **Stage 2: Bayesian Evidence Accumulation on Agent500.** Accuracy averaged over four proxies and five prompt folds after averaging three grouping seeds. All rules reuse the same DC–AC source heads, held-out responses, and query groups.

Accumulation	**Acc.@1**	**Acc.@5**	**Acc.@10**	**Acc.@20**	**Acc.@50**	**Acc.@100**
Majority voting	0.476	0.644	0.749	0.821	0.887	0.923
Probability averaging	0.476	0.723	0.807	0.863	0.915	0.943
BEA	0.476	**0.768**	**0.844**	**0.890**	**0.933**	**0.956**

To assess this structure independently of the projection, we measure neighborhoods in the original cosine space. Across eight proxies, median Top-1/Top-5 family purity is $84.5\%/56.1\%$ against an 18.4% random expectation. Enrichment is significant for every main proxy after Holm correction ($p \leq 8.0 \times 10^{-4}$). The main proxies' distance matrices also agree (median Spearman 0.818). For example, DeepSeek-R1-Distill-Qwen-32B and QwQ-32B are mutual Top-10 neighbors under all eight proxies.

We interpret proximity only as behavioral similarity on Agent500. Establishing model ancestry or shared training history requires independent evidence. Appendix H provides the purity curves, tests in the original space, and stability checks across prompt subsets, projections, and proxy readers.

5 Conclusion

In this paper, we present READER for dynamic LLM provenance from query-varying interactions. Using a frozen proxy, READER constructs comparable spectral fingerprints and combines their source evidence through Bayesian accumulation. Experiments across proxy families demonstrate strong cumulative attribution, while component analyses reveal complementary signals for source identity and model relationships. Together, these findings connect dynamic attribution and static relationship auditing within a shared measurement space. Once the candidate ecosystem is enrolled, ordinary interactions become a growing audit trail.

References

Trenton Bricken, Adly Templeton, Joshua Batson, Brian Chen, Adam Jermyn, Tom Conerly, Nick Turner, Cem Anil, Carson Denison, Amanda Askell, Robert Lasenby, Yifan Wu, Shauna Kravec, Nicholas Schiefer, Tim Maxwell, et al. Towards monosemanticity: Decomposing language models with dictionary learning. Transformer Circuits Thread, https://transformer-circuits.pub/2023/monosemantic-features/, 2023.

Nelson Elhage, Tristan Hume, Catherine Olsson, Nicholas Schiefer, Tom Henighan, Shauna Kravec, Zac Hatfield-Dodds, Robert Lasenby, Dawn Drain, Carol Chen, Roger Grosse, Sam McCandlish,

Jared Kaplan, Dario Amodei, Martin Wattenberg, and Christopher Olah. Toy models of superposition. Transformer Circuits Thread, https://transformer-circuits.pub/2022/toy_model/, 2022.

Thibaud Gloaguen, Robin Staab, Nikola Jovanović, and Martin Vechev. LLM fingerprinting via semantically conditioned watermarks. In *The Fourteenth International Conference on Learning Representations*, 2026.

Martin Gubri, Dennis Ulmer, Hwaran Lee, Sangdoo Yun, and Seong Joon Oh. TRAP: Targeted random adversarial prompt honeypot for black-box identification. In *Findings of the Association for Computational Linguistics: ACL 2024*, pp. 11496–11517, 2024.

John Hewitt and Percy Liang. Designing and interpreting probes with control tasks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*, pp. 2733–2743, 2019.

Yibo Jiang, Goutham Rajendran, Pradeep Kumar Ravikumar, Bryon Aragam, and Victor Veitch. On the origins of linear representations in large language models. In *Proceedings of the 41st International Conference on Machine Learning*, pp. 21879–21911, 2024.

John Kirchenbauer, Jonas Geiping, Yuxin Wen, Jonathan Katz, Ian Miers, and Tom Goldstein. A watermark for large language models. In *International Conference on Machine Learning*, pp. 17061–17084. PMLR, 2023.

Simon Kornblith, Mohammad Norouzi, Honglak Lee, and Geoffrey Hinton. Similarity of neural network representations revisited. In *Proceedings of the 36th International Conference on Machine Learning*, pp. 3519–3529, 2019.

Rohith Kuditipudi, John Thickstun, Tatsunori Hashimoto, and Percy Liang. Robust distortion-free watermarks for language models. *Transactions on Machine Learning Research*, 2024.

Wen Lai, Viktor Hangya, and Alexander Fraser. Style-specific neurons for steering LLMs in text style transfer. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 13427–13443, 2024.

Hope McGovern, Rickard Stureborg, Yoshi Suhara, and Dimitris Alikaniotis. Your large language models are leaving fingerprints. In *Proceedings of the 1st Workshop on GenAI Content Detection*, pp. 85–95, 2025.

Anshul Nasery, Jonathan Hayase, Creston Brooks, Peiyao Sheng, Himanshu Tyagi, Pramod Viswanath, and Sewoong Oh. Scalable fingerprinting of large language models. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025.

Ivica Nikolic, Teodora Baluta, and Prateek Saxena. Model provenance testing for large language models. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025.

Kiho Park, Yo Joong Choe, and Victor Veitch. The linear representation hypothesis and the geometry of large language models. In *Proceedings of the 41st International Conference on Machine Learning*, pp. 39643–39666, 2024.

Dario Pasquini, Evgenios M. Kornaropoulos, and Giuseppe Ateniese. LLMmap: Fingerprinting for large language models. In *34th USENIX Security Symposium*, pp. 299–318, 2025.

Wenjun Peng, Jingwei Yi, Fangzhao Wu, Shangxi Wu, Bin Bin Zhu, Lingjuan Lyu, Binxing Jiao, Tong Xu, Guangzhong Sun, and Xing Xie. Are you copying my model? Protecting the copyright of large language models for EaaS via backdoor watermark. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics*, pp. 7653–7668, 2023.

Xiaoqi Qiu, Hao Zeng, Zhiyu Hou, and Hongxin Wei. Provable model provenance set for large language models. arXiv preprint arXiv:2602.00772, 2026.

Abhilasha Ravichander, Yonatan Belinkov, and Eduard H. Hovy. Probing the probing paradigm: Does probing accuracy entail task relevance? In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics*, pp. 3363–3377, 2021.

Nina Rimsky, Nick Gabrieli, Julian Schulz, Meg Tong, Evan Hubinger, and Alexander Turner. Steering Llama 2 via contrastive activation addition. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics*, pp. 15504–15522, 2024.

Ala N. Tak, Amin Banayeeanzade, Anahita Bolourani, Mina Kian, Robin Jia, and Jonathan Gratch. Mechanistic interpretability of emotion inference in large language models. In *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 13090–13120, 2025.

Paul Tschisgale and Peter Wulff. Evidence for daily and weekly periodic variability in GPT-4o performance. arXiv preprint arXiv:2602.15889, 2026.

Yubo Wang, Xueguang Ma, Ge Zhang, Yuansheng Ni, Abhranil Chandra, Shiguang Guo, Weiming Ren, Aaran Arulraj, Xuan He, Ziyan Jiang, Tianle Li, Max Ku, Kai Wang, Alex Zhuang, Rongqi Fan, Xiang Yue, and Wenhu Chen. MMLU-Pro: A more robust and challenging multi-task language understanding benchmark. In *Advances in Neural Information Processing Systems*, volume 37, pp. 95266–95290, 2024. doi: 10.52202/079017-3018.

Zehao Wu, Yanjie Zhao, and Haoyu Wang. TensorGuard: Gradient-based model fingerprinting for LLM similarity detection and family classification. In *Proceedings of the 40th IEEE/ACM International Conference on Automated Software Engineering*, pp. 445–456, 2025. URL https://conf.researchr.org/details/ase-2025/ase-2025-papers/18/TensorGuard-Gradient-Based-Model-Fingerprinting-for-LLM-Similarity-Detection-and-Fam.

Zhaomin Wu, Haodong Zhao, Ziyang Wang, Jizhou Guo, Qian Wang, and Bingsheng He. LLM DNA: Tracing model evolution via functional representations. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/pdf?id=UIxHaAqFqQ.

Jiashu Xu, Fei Wang, Mingyu Ma, Pang Wei Koh, Chaowei Xiao, and Muhao Chen. Instructional fingerprinting of large language models. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 3277–3306, 2024.

Shojiro Yamabe, Futa Kai Waseda, Tsubasa Takahashi, and Koki Wataoka. MergePrint: Merge-resistant fingerprints for robust black-box ownership verification of large language models. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics*, pp. 6894–6916, 2025.

Xiaofang Yang, Lijun Li, Heng Zhou, Tong Zhu, Xiaoye Qu, Yuchen Fan, Qianshan Wei, Rui Ye, Li Kang, Yiran Qin, et al. Toward efficient agents: Memory, tool learning, and planning. arXiv preprint arXiv:2601.14192, 2026.

Zhiguang Yang and Hanzhou Wu. Unveiling hidden model fingerprints in API-protected LLMs. In *IS&T International Symposium on Electronic Imaging 2026: Media Watermarking, Security, and Forensics*, pp. 309–1–309–9, 2026. doi: 10.2352/EI.2026.38.4.MWSF-309. URL https://library.imaging.org/ei/articles/38/4/MWSF-309.

Nicolas Yax, Pierre-Yves Oudeyer, and Stefano Palminteri. PhyloLM: Inferring the phylogeny of large language models and predicting their performances in benchmarks. In *The Thirteenth International Conference on Learning Representations*, 2025.

Boyi Zeng, Lizheng Wang, Yuncong Hu, Yi Xu, Chenghu Zhou, Xinbing Wang, Yu Yu, and Zhouhan Lin. HuRef: Human-readable fingerprint for large language models. In *Advances in Neural Information Processing Systems*, volume 37, pp. 126332–126362, 2024.

Hengyuan Zhang, Zhihao Zhang, Ercong Nie, Mingyang Wang, Zunhai Su, Yiwei Wang, Qianli Wang, Shuzhou Yuan, Xufeng Duan, Qibo Xue, Zeping Yu, Chenming Shang, Xiao Liang, Jing Xiong, Hui Shen, Chaofan Tao, Zhengwu Liu, Senjie Jin, Zhiheng Xi, Dongdong Zhang, Sophia Ananiadou, Tao Gui, Ruobing Xie, Hayden Kwok-Hay So, Hinrich Schuetze, Xuanjing Huang, Qi Zhang, and Ngai Wong. Locate, steer, and improve: A practical survey of actionable mechanistic interpretability in large language models. In *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 10317–10362, 2026. doi: 10.18653/v1/2026.findings-acl.502. URL https://aclanthology.org/2026.findings-acl.502/.

Jie Zhang, Dongrui Liu, Chen Qian, Linfeng Zhang, Yong Liu, Yu Qiao, and Jing Shao. REEF: Representation encoding fingerprints for large language models. In *The Thirteenth International Conference on Learning Representations*, 2025.

Andy Zou, Long Phan, Sarah Chen, James Campbell, Phillip Guo, Richard Ren, Alexander Pan, Xuwang Yin, Mantas Mazeika, Ann-Kathrin Dombrowski, et al. Representation engineering: A top-down approach to AI transparency. arXiv preprint arXiv:2310.01405, 2023.

Table 6: **Composition of the 500 Agent500 prompts.** Counts cover the entire corpus. Examples are lightly typeset versions of prompts used during collection. Category metadata is used only to describe the corpus.

Prompt category	Count	Representative prompt
Code modification and refactoring	114	Add graceful shutdown handling to the HTTP server defined in `cmd/server/main.go`.
Documentation and communication	87	Draft a PR description for a change that adds rate limiting to our public API endpoints.
Testing and quality assurance	85	Run the failing tests in isolation and summarize what each assertion expected versus received.
Code review and repository analysis	46	Search the codebase for every place that hits the `/api/payments` endpoint and list the call sites.
Data systems and storage	42	Given a 2GB CSV of user events on disk, compute the ten most active users without loading it all into memory.
Infrastructure and cloud operations	41	Detect orphaned EBS volumes across all AWS accounts in our organization and write a script for it.
Observability, performance, and reliability	33	Our Grafana dashboard shows p99 latency spiking at 14:00 UTC daily. Which tools would you use to correlate this with a cause?
Security and compliance	24	I pushed a secret to git three commits ago. Walk me through the safe removal procedure.
Build, CI/CD, and release engineering	21	A user reports that `npm run build` fails with exit code 137. Give a diagnosis plan before touching any files.
API, integration, and automation	7	Translate a REST endpoint into an equivalent GraphQL query and mutation pair.

A Data and Evaluation Protocol

A.1 Agent500 Construction

Agent500 crosses 500 agent-style prompts with a fixed ecosystem of 100 target models, yielding 50,000 prompt–response records. The ecosystem contains 50 locally hosted models and 50 models accessed through APIs, with families spanning Qwen, Llama, Mistral, DeepSeek, Gemma/Gemini, GPT, Claude, GLM, MiniMax, and several independent releases. Table 8 provides the complete roster. Each prompt is issued once to every source. Generation uses the common settings reported in the main paper: temperature 0.7 and top-p 0.95.

The benchmark is balanced by construction: every source contributes exactly 500 responses and every prompt contributes exactly 100 source responses. Source labels identify the served model endpoint, including versioned endpoints when the provider exposes one. Each served, versioned endpoint defines one attribution class. Family metadata is used only for grouped error and geometry analyses.

Prompt composition. The prompts center on software-engineering and operational workflows. They include requests to inspect, modify, test, deploy, diagnose, and document software systems. Targets receive the prompt text through their chat interfaces. Repository contents, tool outputs, and execution state are absent from collection. The resulting records capture source behavior on agent-style requests before tool execution.

Table 6 reports the complete category counts and one recorded example per category. The category labels serve as descriptive corpus metadata. Attribution models receive unannotated prompt–response records. The full prompt list is part of the released benchmark.

A.2 Prompt-Grouped Evaluation

All model selection and evaluation is prompt grouped. A five-fold split with seed 42 partitions the 500 prompt identifiers, so all 100 responses elicited by one prompt remain together. In each outer fold, the other 400 prompts provide 40,000 enrollment records and the held-out 100 prompts provide 10,000 single-response test records. Coordinate standardization, probe fitting, and class-prior estimation use the enrollment records.

At budget $K > 1$, test records are grouped by source into disjoint sets of K responses. We use grouping seeds 42, 43, and 44 for every $K \in \{1, 5, 10, 20, 50, 100\}$. Every budget reuses the same response-level probe. Grouping determines which held-out measurements enter the evidence sum.

For $K = 1$, the grouping seeds induce the same set of 50,000 out-of-fold predictions, making the score deterministic across seeds.

Metric reduction. The main endpoint tables report means over the five outer folds. Full budget curves pool all out-of-fold predictions before computing each metric and then average over the three grouping seeds. Accuracy is linear in the examples and is identical under these two reductions. Macro-F1 is nonlinear, so pooled values can differ slightly from the mean of fold-wise macro-F1. Macro-F1 first computes one-vs-rest F1 for each of the 100 sources and then averages the 100 values.

A.3 Proxy Readers and Layer Selection

The prompt and response are jointly processed by each proxy (the `ur` input view). The DCT receives the hidden states aligned with response tokens. For each outer fold, its 400 enrollment prompts retain their four prompt-fold assignments as inner folds. Every Transformer block is a candidate layer. At each layer, we fit the coordinate standardizer and response-level source probe on three inner folds and measure accuracy on the fourth, rotating through all four held-out folds. We select the layer with the highest mean inner-validation accuracy. Layer scores are compared at full numerical precision. Exact ties favor the shallower layer. We refit the standardizer and probe at the selected layer on all 40,000 enrollment records and evaluate once on the untouched outer fold. The resulting fold-local layer and probe are reused for every query budget and the associated shifted evaluations. The matched representation controls in Section 4.5 use the fixed task presets detailed in Appendix G.1. The first four proxies constitute the main roster. The supplement reports all eight.

For the capability diagnostic in Appendix E.3, MMLU-Pro (Wang et al., 2024) uses category-matched 5-shot chain-of-thought demonstrations, temperature zero, a 4000-token output limit, and disabled thinking mode. These capability scores serve only the proxy diagnostic.

A.4 Shifted Evaluation Protocols

The controlled-length test truncates Agent500 responses to the first 32 or 64 target-generation tokens before proxy extraction. The truncated versions retain the original prompt identifiers, allowing each record to be evaluated by the outer-fold model that excluded its prompt during training. Math100 crosses 100 mathematical problems with the same 100 candidate sources, yielding 10,000 new records. Each of the five frozen Agent500 fold models evaluates all Math100 responses independently. For each fold model and grouping seed, we accumulate response-level evidence at the same six query budgets. Tables report the mean of the five fold-model metrics. Full budget curves additionally average over the three grouping seeds.

Both tests reuse the complete clean-Agent500 estimator, including its proxy layer, coordinate standardizer, source probe, and class prior. All layer selection, standardization, probe fitting, and prior estimation use Agent500 enrollment records exclusively. The resulting scores measure transfer from the enrolled system without retraining.

Math100 construction. Math100 is a separately constructed set of 100 proof-oriented and derivational mathematics prompts. It spans real and complex analysis, algebra, topology, differential equations, probability, number theory, combinatorics, optimization, numerical analysis, and functional analysis. Every source receives the same 100 prompt texts under the Agent500 collection convention. Evaluation scores source identity for each response. Table 7 shows representative prompts. The domain labels provide descriptive summaries of these examples.

A.5 Static Relationship Evaluation on Bench-A

Starting from Bench-A, introduced with MPT, we construct a balanced binary relationship classification task with 116 pairs spanning 67 unique models. The models form nine *parent groups*, each consisting of one designated parent checkpoint and its derivatives listed in the benchmark. Group membership is defined by the exact parent checkpoint. Models from different groups may share a provider or architecture. Positive examples pair a parent checkpoint with a derived model. Negative examples pair models from different parent groups. Every model contributes 600 prompt-aligned responses. Appendix F.2 provides the complete model and pair roster for pair holdout. For each pair

Table 7: **Representative prompts from Math100.** The evaluation of domain shift without retraining uses the complete 100-prompt collection.

Illustrative domain	Representative Math100 prompt
Real analysis	Determine, with proof, whether $\sum_{n=2}^{\infty}[n(\ln n)^p]^{-1}$ converges as a function of $p > 0$.
Abstract algebra	Let G be a finite group of order p^2, where p is prime. Prove that G is abelian.
Topology	Show that the fundamental group of S^1 is isomorphic to $\mathbb{Z}$ using the universal cover $\mathbb{R} \to S^1$.
Probability	A random walk on $\mathbb{Z}$ starts at zero and moves $+1$ with probability p. Compute the probability that it ever reaches $+1$.
Number theory	Prove that an integer is a sum of two squares exactly when every prime factor congruent to 3 modulo 4 has even exponent.
Optimization	For a strictly convex differentiable function with Lipschitz gradient, prove that gradient descent with step size $1/L$ converges at rate $O(1/k)$.

and aligned prompt, READER computes DC and first AC cosine similarities separately, averages each score over the 600 prompts, and trains a linear SVM on the resulting two-dimensional feature. For each proxy, Bench-A uses the modal layer among the five Agent500 outer-fold selections. A modal tie is resolved by mean inner-validation accuracy across the five Agent500 outer folds and then in favor of the shallower layer. Agent500 therefore determines the proxy layer, while Bench-A labels train only the pair readout over two cosine scores. The scores are standardized within each training split. The split is applied at the pair level, and the readout is fitted from the derived relationship labels.

Pair holdout. The default protocol in the main paper uses the same 20 fixed stratified $4{:}1$ splits for every method. Each split assigns 92 model pairs to training and 24 to testing. Training and test pairs are distinct, while individual models may recur across the two sets.

Model holdout. Each of 20 splits holds out the parent checkpoint and at least two derivatives from each of two parent groups. Other derivatives from those groups appear in training, so the parent groups overlap. All test models, including the two parent checkpoints, are absent from training.

Parent group holdout. We evaluate all $\binom{9}{2} = 36$ ways to hold out two complete parent groups, including their parent checkpoints and all their derivatives. The training and test sets contain distinct models and parent groups. Providers and architectures may occur in both sets.

For model holdout and parent group holdout, positive pairs connect parents to their derivatives, and negatives are sampled uniformly without replacement from pairs belonging to different parent groups. Training and test sets are balanced independently within every split.

B Implementation and Baselines

B.1 READER Optimization

Each fold-local source probe is a full-batch multinomial logistic regressor. The coordinate standardizer uses $g_\psi(\mathbf{u}) = (\mathbf{u} - \boldsymbol{\mu}) \oslash \boldsymbol{\sigma}$, where $\psi = (\boldsymbol{\mu}, \boldsymbol{\sigma})$ contains enrollment-fold means and scales for normalization to unit variance, and $\oslash$ denotes elementwise division. We then optimize cross-entropy with Adam for 40 steps at learning rate 10^{-3}. The learning rate follows cosine decay toward one percent of its initial value over a 100-step horizon. The objective applies L2 regularization to the weight matrix, with coefficient $\lambda = N_{\mathrm{tr}}^{-1}$ in the main-text objective, and leaves the bias unregularized. Proxy weights are frozen. The trainable state in each fold comprises the standardizer and $2d \times 100$ linear source head.

B.2 Dynamic Baselines

The dynamic baselines use the same prompt–response (`ur`) records as the primary READER evaluation.

Fine-tuned DeBERTa. DeBERTa-v3-large is fine-tuned end to end as a 100-way text classifier under the same five prompt-grouped folds. Each prompt–response record is serialized as the classi-

Algorithm 1 Spectral Trajectory Encoding

Require: Prompt p, response x, frozen proxy ϕ, proxy layer ℓ
Ensure: Response fingerprint $\mathbf{u} \in \mathbb{R}^{2d}$

1: Run ϕ on (p, x) and retain the layer-ℓ states aligned with the T response tokens, $\{\mathbf{h}_t\}_{t=0}^{T-1}$
2: $\mathbf{z}_0 \leftarrow \dfrac{1}{T}\displaystyle\sum_{t=0}^{T-1} \mathbf{h}_t$
3: $\mathbf{z}_1 \leftarrow \dfrac{\sqrt{2}}{T}\displaystyle\sum_{t=0}^{T-1} \mathbf{h}_t \cos\left(\dfrac{\pi(t+1/2)}{T}\right)$
4: $\mathbf{u} \leftarrow [\mathbf{z}_0^\top, \mathbf{z}_1^\top]^\top$
5: **return** $\mathbf{u}$

fier input and truncated to 256 tokenizer tokens. Each fold is trained for two epochs with AdamW, learning rate 6×10^{-6}, weight decay 0.01, and a linear warmup ratio of 0.06 in FP32. At $K > 1$, its response-level log-posteriors are accumulated with the same rule as READER. This gives the text classifier the identical test grouping and query budgets.

LLMmap and LLM-DNA. The published LLMmap and LLM-DNA evaluations rely on a controlled panel in which candidates answer a fixed shared prompt set. We adapt both methods to Agent500's query-varying interactions on unseen prompts. LLMmap's closed-set behavioral score is applied to each observed prompt–response record before query grouping. For LLM-DNA, we freeze MPNet, BGE-large, or Qwen3-Embedding-8B, embed each prompt–response record, and fit a record-level source probe using the same prompt-grouped training folds. The resulting posteriors use the same evidence accumulation and grouping as READER. This matched protocol isolates the interaction representation under a common dynamic decision rule.

B.3 Static Baselines

The Bench-A-derived comparison uses each method's native static relationship pipeline. MPT and PhyloLM operate on their pairwise behavioral features. LLM-DNA follows its original concat–projection–centroid procedure. READER uses the two averaged cosine scores defined in Section A. Every method receives the same 20 splits for pair holdout. Preprocessing and supervised readouts are fitted independently within each split.

Model holdout and parent group holdout reuse these cached method-native pair features. MPT uses first-token agreement, PhyloLM uses first-four-character agreement, and the three LLM-DNA variants use Euclidean distance between their fixed GRP-128 model vectors. For every method and split, a linear SVM and its scalar-feature standardizer are fitted from training pairs. Thus these protocols isolate identity overlap under a fixed representation and readout class.

C Agent500 Target Roster

Table 8 lists all 100 enrolled sources. Entries 1–50 were locally hosted checkpoints. Entries 51–100 were served API endpoints. We preserve version suffixes because endpoint versions are distinct candidate sources in the closed ecosystem. The roster includes closely related variants, making within-family distinctions the hardest and most operationally relevant part of the 100-way task.

The *Access* field records the response-collection route. *Family group* provides the coarse organization used for confusion matrices and ecosystem geometry. Exact architecture and training lineage are unannotated.

D READER Inference

Algorithm 1 implements Spectral Trajectory Encoding. Algorithm 2 applies Bayesian Evidence Accumulation to an arbitrary query budget using a single response-level probe.

Table 8: **The 100 Agent500 candidate sources.** Model identifiers are retained exactly as recorded in the validated experiment manifest. Access denotes the collection route. Family groups provide coarse analysis labels.

Locally hosted sources				API sources			
ID	**Model or endpoint identifier**	**Access**	**Family group**	**ID**	**Model or endpoint identifier**	**Access**	**Family group**
1	Qwen/Qwen2.5-3B-Instruct	Local	Qwen	51	gpt-4o	API	GPT
2	Qwen/Qwen2.5-7B-Instruct	Local	Qwen	52	gpt-4o-2024-11-20	API	GPT
3	Qwen/Qwen2.5-14B-Instruct	Local	Qwen	53	gpt-4.1-mini-2025-04-14	API	GPT
4	Qwen/Qwen2.5-32B-Instruct	Local	Qwen	54	gpt-4.1-2025-04-14	API	GPT
5	Qwen/Qwen2.5-72B-Instruct	Local	Qwen	55	gpt-4.1-nano-2025-04-14	API	GPT
6	Qwen/Qwen3-1.7B	Local	Qwen	56	gpt-5.4-nano	API	GPT
7	Qwen/Qwen3-1.7B-Base	Local	Qwen	57	qwen3.5-flash	API	Qwen
8	Qwen/Qwen3-4B-Instruct-2507	Local	Qwen	58	qwen3.5-plus	API	Qwen
9	Qwen/Qwen3-4B-Thinking-2507	Local	Qwen	59	gpt-5.4-mini	API	GPT
10	Qwen/Qwen3-14B	Local	Qwen	60	gpt-5.4	API	GPT
11	Qwen/Qwen3-30B-A3B-Instruct-2507	Local	Qwen	61	gpt-5.5	API	GPT
12	Qwen/Qwen3-30B-A3B-Thinking-2507	Local	Qwen	62	gpt-5.6-luna	API	GPT
13	Qwen/Qwen3-Next-80B-A3B-Instruct	Local	Qwen	63	gpt-5.6-sol	API	GPT
14	Qwen/Qwen3-Next-80B-A3B-Thinking	Local	Qwen	64	gpt-5.6-terra	API	GPT
15	Qwen/Qwen3-Coder-30B-A3B-Instruct	Local	Qwen	65	qwen3.6-flash	API	Qwen
16	Qwen/QwQ-32B	Local	Qwen	66	qwen3.6-plus	API	Qwen
17	Qwen/Qwen3.5-4B	Local	Qwen	67	glm-4.7-FlashX	API	GLM
18	Qwen/Qwen3.5-4B-Base	Local	Qwen	68	claude-opus-4-6	API	Claude
19	Qwen/Qwen3.5-9B	Local	Qwen	69	claude-opus-4-7	API	Claude
20	Qwen/Qwen3.5-9B-Base	Local	Qwen	70	claude-opus-4-8	API	Claude
21	Qwen/Qwen3.5-27B	Local	Qwen	71	claude-sonnet-4-20250514	API	Claude
22	Qwen/Qwen3.5-122B-A10B	Local	Qwen	72	glm-4.6	API	GLM
23	Qwen/Qwen3.6-27B	Local	Qwen	73	claude-sonnet-4-6	API	Claude
24	Qwen/Qwen1.5-MoE-A2.7B-Chat	Local	Qwen	74	mistralai/mistral-small-24b-instruct-2501	API	Mistral
25	meta-llama/Meta-Llama-3-8B-Instruct	Local	Llama	75	gemini-2.5-flash-lite	API	Gemini
26	meta-llama/Meta-Llama-3-70B-Instruct	Local	Llama	76	deepseek-v4-flash	API	DeepSeek
27	meta-llama/Llama-3.1-8B-Instruct	Local	Llama	77	qwen3.6-max-preview	API	Qwen
28	meta-llama/Llama-3.1-70B-Instruct	Local	Llama	78	gemini-3.1-flash-lite-preview	API	Gemini
29	meta-llama/Llama-3.2-1B-Instruct	Local	Llama	79	qwen3.7-plus	API	Qwen
30	meta-llama/Llama-3.2-3B-Instruct	Local	Llama	80	mistralai/mistral-medium-3	API	Mistral
31	meta-llama/Llama-4-Scout-17B-16E-Instruct	Local	Llama	81	deepseek-ai/DeepSeek-V3	API	DeepSeek
32	mistralai/Mistral-7B-Instruct-v0.3	Local	Mistral	82	deepseek-ai/DeepSeek-V3.1-Terminus	API	DeepSeek
33	mistralai/Mistral-Nemo-Instruct-2407	Local	Mistral	83	deepseek-ai/DeepSeek-V3.2	API	DeepSeek
34	mistralai/Mixtral-8x7B-Instruct-v0.1	Local	Mistral	84	deepseek-v4-pro	API	DeepSeek
35	deepseek-ai/DeepSeek-R1-Distill-Qwen-1.5B	Local	DeepSeek	85	qwen3-coder-480b-a35b-instruct	API	Qwen
36	deepseek-ai/DeepSeek-R1-Distill-Qwen-7B	Local	DeepSeek	86	qwen3-max-2026-01-23	API	Qwen
37	deepseek-ai/DeepSeek-R1-Distill-Qwen-14B	Local	DeepSeek	87	qwen3.5-397b-a17b	API	Qwen
38	deepseek-ai/DeepSeek-R1-Distill-Qwen-32B	Local	DeepSeek	88	qwen3.7-max-2026-05-20	API	Qwen
39	deepseek-ai/DeepSeek-V2-Lite-Chat	Local	DeepSeek	89	MiniMax-Text-01	API	MiniMax
40	google/gemma-3-27b-it	Local	Gemma	90	mistralai/mistral-small-3.1-24b-instruct	API	Mistral
41	google/gemma-4-26B-A4B-it	Local	Gemma	91	MiniMax-M3	API	MiniMax
42	google/gemma-4-31B-it	Local	Gemma	92	glm-4.7	API	GLM
43	microsoft/Phi-3-mini-4k-instruct	Local	Phi	93	glm-5.1	API	GLM
44	tencent/Hunyuan-1.8B-Instruct	Local	Hunyuan	94	glm-5.2	API	GLM
45	openai/gpt-oss-20b	Local	GPT	95	mistralai/mistral-medium-3-5	API	Mistral
46	openai/gpt-oss-120b	Local	GPT	96	kimi-k3	API	Kimi
47	zai-org/GLM-4.5-Air	Local	GLM	97	mistralai/mistral-large-2512	API	Mistral
48	ByteDance-Seed/Seed-OSS-36B-Instruct	Local	Seed	98	mistralai/mistral-medium-3.1	API	Mistral
49	baidu/ERNIE-4.5-21B-A3B-Thinking	Local	ERNIE	99	meta-llama/llama-3.3-70b-instruct	API	Llama
50	LiquidAI/LFM2-24B-A2B	Local	LFM	100	meta-llama/llama-4-maverick	API	Llama

Algorithm 2 Bayesian Evidence Accumulation

Require: Observations $\{(p_k, x_k)\}_{k=1}^{K}$, frozen proxy ϕ, layer ℓ, train-fitted standardizer g_ψ, source probe q_θ, class prior $\boldsymbol{\pi}$, floor $\varepsilon > 0$

Ensure: Predicted source $\widehat{y}$

for $c = 1$ **to** C **do**
  $S_c \leftarrow -\dfrac{K-1}{K} \log \pi_c$
end for
for $k = 1$ **to** K **do**
  $\mathbf{u}_k \leftarrow \mathrm{SpectralEncode}(p_k, x_k; \phi, \ell)$
  $\widetilde{\mathbf{u}}_k \leftarrow g_\psi(\mathbf{u}_k)$
  $\mathbf{q}_k \leftarrow \mathrm{softmax}(\mathbf{W}^\top \widetilde{\mathbf{u}}_k + \mathbf{b})$
  for $c = 1$ **to** C **do**
    $S_c \leftarrow S_c + \dfrac{1}{K} \log(q_{k,c} + \varepsilon)$
  end for
end for
$\widehat{y} \leftarrow \arg\max_{c \in \mathcal{C}} S_c$
return $\widehat{y}$

D.1 Prior Correction

The accumulation rule follows directly from the stateless query model. For fixed observed prompts $p_{1:K}$ and source prior π_c, conditional independence gives

$$p(c \mid x_{1:K}, p_{1:K}) \propto \pi_c \prod_{k=1}^{K} p(x_k \mid p_k, c). \tag{11}$$

Bayes' rule for one response gives

$$p(x_k \mid p_k, c) = \frac{p(c \mid p_k, x_k)\, p(x_k \mid p_k)}{\pi_c}, \tag{12}$$

where prompt selection is source independent and therefore uses the same π_c for every request. Substituting Eq. 12 into Eq. 11, removing terms independent of c, and replacing the single-response posterior by the probe q_θ yields

$$\log p(c \mid x_{1:K}, p_{1:K}) \doteq \sum_{k=1}^{K} \log q_\theta(c \mid \mathbf{u}_k) - (K-1) \log \pi_c, \tag{13}$$

where $\doteq$ denotes equality up to a class-independent constant. Dividing by K preserves the ranking and produces the score used in the paper. This normalization keeps score magnitudes comparable across query budgets. In the balanced Agent500 ecosystem, $\pi_c = 1/C$, so the correction is constant and the classifier reduces to mean log-posterior accumulation. The $1/K$ normalization preserves class rankings while leaving the softmax scale budget dependent. We use the accumulated outputs as decision scores. Budget-specific held-out calibration is required for a probabilistic interpretation.

The factorization applies to request state and permits prompts from different tasks and domains. It is exact for independent, stateless generations conditioned on their source and observed prompts. Shared hidden conversational state turns Eq. 13 into a composite evidence score. A state-aware model would be required to calibrate that setting. Agent500 uses independent requests.

D.2 Dimensions and Cost

For proxy hidden size d, one response produces two d-dimensional coefficients and a $2d$-dimensional fingerprint. Computing both coefficients requires one pass over the T response states, $O(Td)$ arithmetic, and can be implemented without materializing a $T \times T$ transform. The dense C-source probe costs $O(Cd)$ per response, and accumulation costs $O(KC)$ for a group of K observations. Stored fingerprints require $2d$ values per record. Once the target response is observed, the audit invokes the frozen proxy, which dominates its computational cost.

E Complete Attribution Results

E.1 Endpoint Attribution and Computational Cost

Table 9 reports endpoint attribution and computational cost for all eight proxy readers and the dynamic baselines.

FT DNA with Qwen3-Emb leads at $K = 1$ (0.554 versus 0.504). At $K = 100$, READER with Qwen3.5-9B reaches 0.962 accuracy and 0.952 macro-F1, reversing the ranking by 11.2 and 14.6 points. READER uses a $54\times$ smaller trainable set (819.3K versus 44.1M) and a $1.55\times$ smaller online path (4.91B versus 7.61B). With cached fingerprints, fitting is roughly $30{,}000\times$ shorter (0.35 seconds versus 173 minutes), with comparable runtime (6.50 versus 6.13 ms per response). FT adaptation favors single-shot classification. READER combines lower adaptation cost with substantially stronger cumulative attribution.

E.2 All Query Budgets and Proxy Readers

Table 10 reports the complete accuracy trajectory for all eight frozen proxies. Figure 5 places the dynamic baselines on the same axes and gives the corresponding macro-F1 curves. Every reader

Table 9: **Agent500 attribution and computational profile.** Attribution results are means over five prompt-grouped folds. Random uses 20,000 Monte Carlo trials at fold-sized test budgets. Computational measurements use one NVIDIA H200 with 140 GB HBM. Trainable parameters and fit time cover one 40,000-response outer-training fold. For READER, fit time starts from cached fingerprints for every candidate layer and includes all inner-fold probe fits, layer selection, and the final outer-training probe fit. Proxy forward passes are reported through audit-time latency. Audit-path parameters include the representation path and 100-way head. Latency is the median of three warmed runs, expressed in seconds per response at each implementation's validated batch size. FT DNA updates the complete MPNet/BGE encoder and uses LoRA for Qwen3-Embedding-8B. †Dynamic adaptation of a fixed-panel method.

Method / backbone	**Attribution at $K = 1$**		**Attribution at $K = 100$**		**Offline fit**		**Online audit**	
	Acc.	**M-F1**	**Acc.**	**M-F1**	**Trainable params.**	**Fit time per fold**	**Audit-path params.**	**Latency (s/resp.)**
Random	0.010	0.010	0.010	0.007	–	–	–	–
FT DeBERTa / DeBERTa-v3-L	0.330	0.295	0.568	0.471	435.2M	7.3 min	435.2M	0.00290
LLMmap† / E5-L	0.212	0.189	0.522	0.423	3.05M	10.3 s	562.9M	0.00061
DNA† / MPNet	0.119	0.109	0.706	0.644	76.9K	0.17 s	109.6M	0.00101
DNA† / BGE-L	0.191	0.188	0.790	0.739	102.5K	0.17 s	335.2M	0.00291
DNA† / Qwen3-Emb-8B	0.185	0.192	0.786	0.735	409.7K	0.41 s	7.57B	0.00613
FT DNA† / MPNet	0.230	0.170	0.310	0.218	109.6M	2.4 min	109.6M	0.00101
FT DNA† / BGE-L	0.303	0.237	0.458	0.347	335.2M	5.3 min	335.2M	0.00291
FT DNA† / Qwen3-Emb-8B	**0.554**	**0.530**	0.850	0.806	44.1M	173 min	7.61B	0.00613
READER / Llama-3.1-8B-Instruct	0.456	0.455	0.948	0.931	819.3K	0.55 s	962.4M	0.00053
READER / Gemma-4-12B	0.460	0.457	0.958	0.945	768.1K	0.63 s	2.13B	0.00123
READER / Ministral-3-8B	0.483	0.481	0.952	0.939	819.3K	0.64 s	4.25B	0.00274
READER / Qwen3.5-9B	0.504	0.503	0.962	0.952	819.3K	0.35 s	4.91B	0.00650
READER / OLMo-3-7B-Instruct	0.463	0.462	0.942	0.930	819.3K	0.35 s	3.25B	0.00247
READER / Qwen3-8B	0.447	0.447	0.948	0.933	819.3K	0.36 s	1.01B	0.00100
READER / Qwen3-32B	0.509	0.508	**0.964**	**0.955**	1.02M	0.43 s	24.18B	0.02141
READER / Qwen3.5-27B	0.523	0.522	0.962	0.952	1.02M	0.42 s	19.92B	0.07903

Table 10: **Agent500 accuracy across all query budgets and proxies.** Mean±standard deviation across three grouping seeds after pooling out-of-fold predictions. This measures grouping sensitivity, not fold variation (Table 1). Grouping does not change the endpoint decisions. Deviations below 0.0005 round to 0.000. The first four rows are the main proxy roster.

Proxy	$K = 1$	$K = 5$	$K = 10$	$K = 20$	$K = 50$	$K = 100$
Llama-3.1-8B-Instruct	0.456±0.000	0.743±0.001	0.826±0.002	0.877±0.002	0.927±0.003	0.948±0.000
Gemma-4-12B-it	0.460±0.000	0.756±0.002	0.834±0.001	0.885±0.002	0.932±0.004	0.958±0.000
Ministral-3-8B	0.483±0.000	0.779±0.003	0.849±0.002	0.893±0.002	0.929±0.000	0.952±0.000
Qwen3.5-9B	0.504±0.000	0.792±0.002	0.865±0.000	0.905±0.004	0.942±0.000	0.962±0.000
OLMo-3-7B-Instruct	0.463±0.000	0.751±0.002	0.828±0.002	0.878±0.003	0.918±0.001	0.942±0.000
Qwen3-8B	0.447±0.000	0.735±0.002	0.816±0.005	0.869±0.003	0.925±0.006	0.948±0.000
Qwen3-32B	0.509±0.000	0.795±0.003	0.865±0.001	0.902±0.001	0.939±0.003	**0.964±0.000**
Qwen3.5-27B	**0.523±0.000**	**0.806±0.001**	**0.873±0.002**	**0.910±0.003**	**0.947±0.001**	0.962±0.000

improves monotonically with additional responses. The spread between proxies is modest relative to the margin over the baselines: at $K = 100$, all eight readers lie between 94.2% and 96.4% accuracy. Qwen3.5-27B gives the strongest single-response result at 52.3%. Qwen3-32B attains the highest Acc.@100 at 96.4%.

E.3 Capability Correlations Across Query Budgets

To inform proxy selection, we examine how general language capability relates to attribution accuracy across the full roster of eight frozen proxies. Figure 6 compares two capability measures with Agent500 Acc.@100. The left panel uses MMLU-Pro, evaluated under the non-thinking 5-shot protocol in Section A, and yields Pearson $r = 0.84$ and Spearman $\rho = 0.81$.

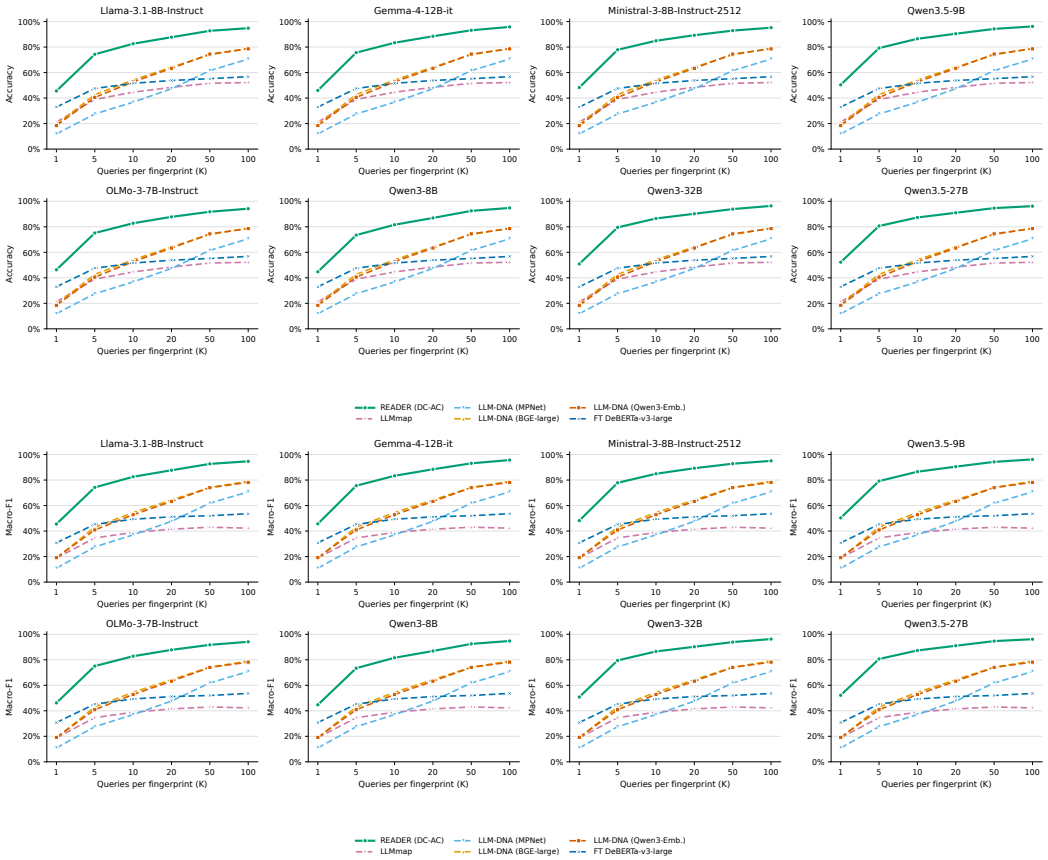


Figure 5: **Complete Agent500 accuracy (top) and macro-F1 (bottom) curves.** Within each outer fold, the enrollment-fitted response-level probe and selected layer are reused across budgets. Baselines use the same held-out prompt groups and evidence budgets. Macro-F1 is computed after pooling the out-of-fold predictions for each grouping seed and therefore differs slightly from the mean of fold-wise macro-F1 in Table 9.

For a second capability measure, we use GPQA-Diamond scores reported by Artificial Analysis.[1] All eight scores use the non-reasoning setting. The right panel yields Pearson $r = 0.78$ and Spearman $\rho = 0.81$ at $K = 100$. MMLU-Pro and GPQA-Diamond induce the same proxy ordering, so their identical Spearman correlations reflect a shared ranking.

Table 11: **Capability–attribution correlations at the endpoint budgets.** Parentheses contain two-sided p-values over the eight proxies. MMLU-Pro and GPQA-Diamond induce the same proxy ranking, so their Spearman correlations are identical at a fixed K.

Budget	**MMLU** r	**GPQA** r	**Shared** ρ
$K = 1$	0.695 (0.056)	0.713 (0.047)	0.714 (0.047)
$K = 100$	0.838 (0.009)	0.779 (0.023)	0.807 (0.015)

Table 11 reports both endpoint budgets. The observed correlations are positive at $K = 1$ and numerically larger at $K = 100$. These results provide an exploratory diagnostic of reader choice. The roster includes four Qwen models, and Acc.@100 occupies a narrow range from 94.2% to 96.4%. A broader comparison controlling for model family and scale would clarify how reliably capability predicts attribution.

[1] https://artificialanalysis.ai/evaluations/gpqa-diamond

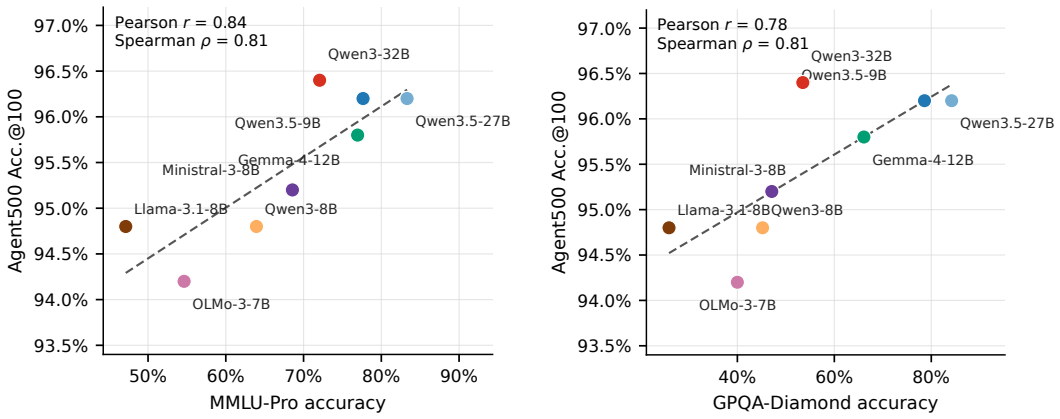


Figure 6: **Proxy capability and attribution at $K = 100$.** The left panel uses our MMLU-Pro scores. The right uses non-reasoning GPQA-Diamond scores from Artificial Analysis. Agent500 accuracy is measured by us, and dashed lines are least-squares fits over the same eight frozen proxies.

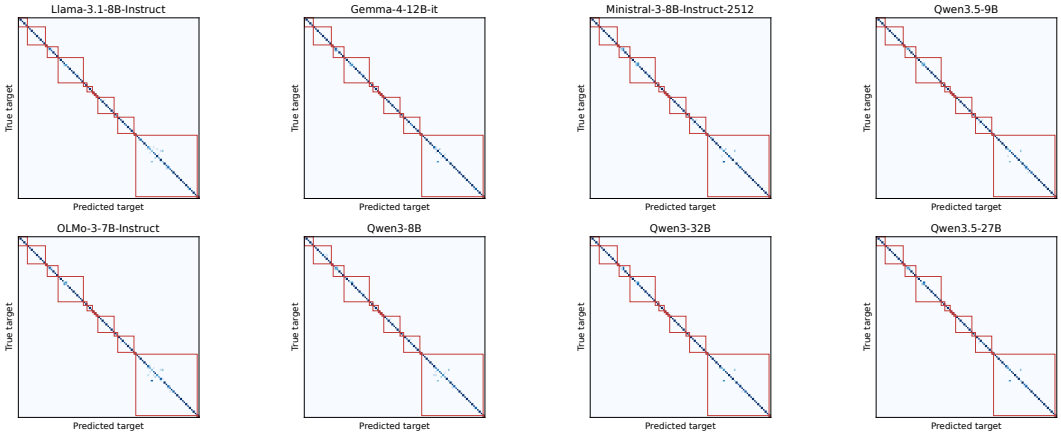


Figure 7: **Out-of-fold confusion at $K = 100$ for all proxy readers.** Matrices are row normalized. Red boxes delimit target families.

E.4 Residual Confusion

Figure 7 reports out-of-fold confusion for all eight proxies. Targets are ordered by provider family. At $K = 100$, errors remain tightly concentrated in the diagonal family blocks across both the main and supplementary readers. The eight readers show qualitatively similar within-family confusion patterns. Closely related endpoint releases and instruction variants account for the residual errors.

E.5 Transfer without Retraining

The following tests hold the complete Agent500 enrollment pipeline fixed while varying the observed responses. Every method reuses its clean fold model, including all preprocessing and source-head parameters.

E.5.1 Controlled Response Length

The response-length columns of Table 2 compare the four main proxies and matched baselines. Truncation causes a substantial absolute decline, particularly at 32 tokens. At 64 tokens, READER with Qwen reaches 50.5% accuracy and READER with Gemma reaches 42.9%, both above fine-tuned DeBERTa's 27.8%. At 32 tokens, READER and DeBERTa are much closer. Severe shortening shifts the decision boundary learned from full responses, locating a clear response-coverage limit.

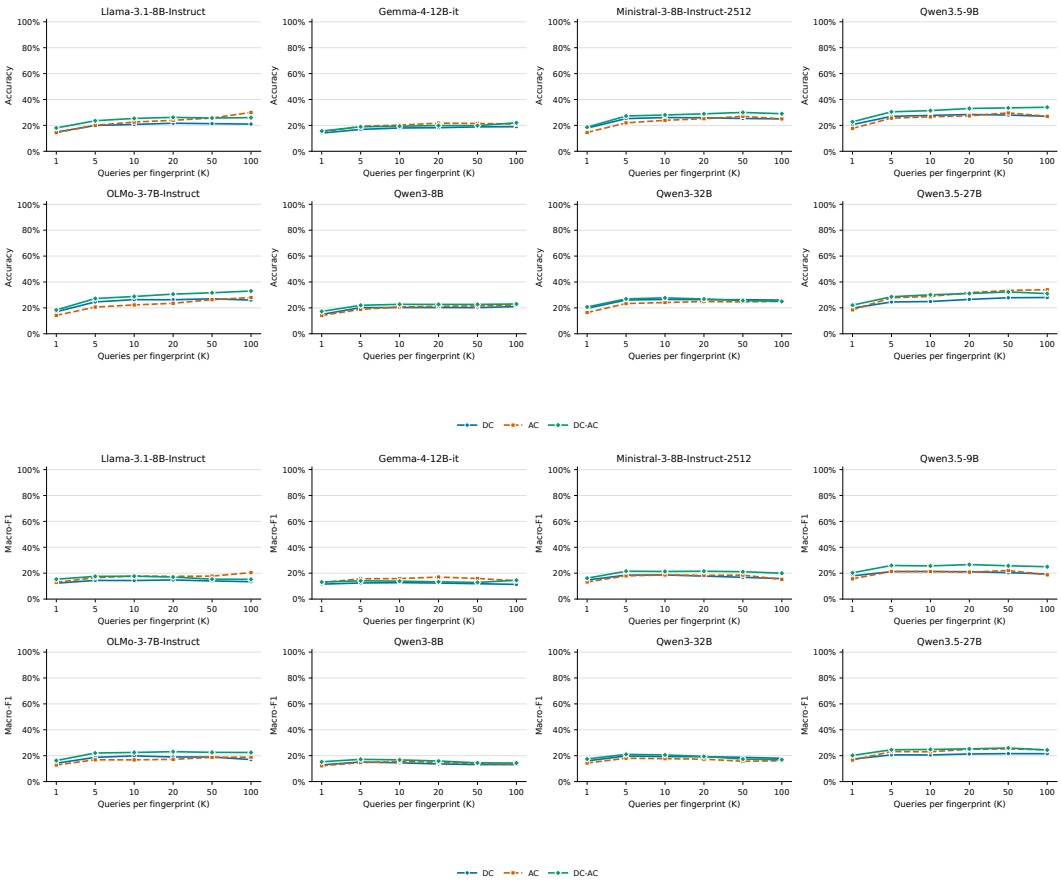


Figure 8: **Math100 accuracy (top) and macro-F1 (bottom) without retraining.** Curves average metrics across the five independently evaluated Agent500 fold models and three grouping seeds. Domain shift depresses macro-F1 more strongly than accuracy, indicating uneven transfer across the 100 candidate sources.

E.5.2 MATH100 DOMAIN SHIFT

The Math100 columns of Table 2 report mean performance across five fold models at $K = 100$. Proxy choice has a larger effect under this domain shift. READER with Qwen and READER with Ministral reach 33.0% and 28.2% accuracy, respectively, whereas READER with Gemma reaches 20.4%. The same separation appears across query budgets in Figure 8. Additional evidence improves both metrics for every reader. The shifted-domain ceiling remains well below in-domain performance, highlighting the value of enrollment coverage over the target prompt distribution.

F COMPLETE STATIC RELATIONSHIP RESULTS

Table 3 reports the pair holdout results. This appendix extends the evaluation to model holdout and parent group holdout and provides the complete model and pair roster.

F.1 MODEL HOLDOUT AND PARENT GROUP HOLDOUT

Table 12 separates pair classification from recurring endpoint identity. Removing model overlap lowers every READER result relative to pair holdout. Under model holdout, Llama and Qwen reach 75.6% and 75.1% accuracy, exceeding PhyloLM by 9.2 and 8.7 points. The corresponding intervals remain wide because each split contains few positive parent–derived edges.

Table 12: **Static relationship transfer under model holdout and parent group holdout.** Means are over 20 splits for model holdout and 36 for parent group holdout. The former permits overlap in parent groups, while the latter holds out two complete groups. Groups are defined by designated parent checkpoints. Accuracy intervals use a percentile bootstrap over splits. Since held-out sets overlap across splits, the intervals and paired tests quantify sensitivity across the specified protocols.

(a) Model holdout

Method	Acc.	Acc. 95% CI	F1	AUC
MPT	0.622	[0.571,0.675]	0.611	0.909
PhyloLM	0.664	[0.615,0.715]	0.623	**0.913**
DNA / MPNet	0.613	[0.563,0.665]	0.589	0.839
DNA / BGE	0.645	[0.554,0.735]	0.673	0.803
DNA / Qwen-Emb.	0.655	[0.598,0.712]	0.612	0.903
READER / Llama	**0.756**	**[0.689,0.822]**	**0.735**	0.895
READER / Gemma	0.657	[0.581,0.732]	0.671	0.854
READER / Ministral	0.699	[0.620,0.778]	0.664	0.864
READER / Qwen	0.751	[0.676,0.821]	0.697	0.856

(b) Parent group holdout

Method	Acc.	Acc. 95% CI	F1	AUC
MPT	0.677	[0.639,0.716]	0.674	0.901
PhyloLM	0.706	[0.661,0.753]	0.683	**0.904**
DNA / MPNet	0.630	[0.597,0.663]	0.617	0.806
DNA / BGE	0.670	[0.632,0.707]	0.678	0.803
DNA / Qwen-Emb.	0.667	[0.636,0.698]	0.640	0.873
READER / Llama	0.663	[0.619,0.707]	0.669	0.825
READER / Gemma	0.624	[0.591,0.656]	0.646	0.772
READER / Ministral	0.663	[0.626,0.701]	0.649	0.813
READER / Qwen	**0.732**	**[0.693,0.771]**	**0.685**	0.830

Holding out complete parent groups imposes a stricter separation. Qwen is strongest at 73.2% accuracy. The other readers fall below PhyloLM. Among the paired comparisons with PhyloLM, Gemma is the only difference that remains significant after Holm correction. It is 8.2 points lower under parent group holdout ($p_{\mathrm{Holm}} = 0.012$). Relationship transfer therefore depends strongly on the parent groups reserved for testing and the proxy reader.

The spectral decomposition also clarifies transfer under parent group holdout. Here, first AC alone exceeds the joint DC–AC readout by 11.1, 8.9, 9.1, and 1.5 accuracy points for Llama, Gemma, Ministral, and Qwen, respectively. DC carries most of the dynamic source signal. In this setting, adding the DC score lowers accuracy by the reported amounts. First AC therefore provides the stronger readout under parent group holdout, while the joint fingerprint remains the default for the primary dynamic task.

Table 13: **Spectral components on the Bench-A-derived pair task.** Mean±standard deviation across four proxy means, each averaged over 20 fixed splits for pair holdout. Deviations measure proxy-to-proxy variation. The last row applies a data-independent Gaussian random projection before cosine similarity.

Pair feature	Acc.	F1	AUC
DC cosine	0.541±0.040	0.636±0.025	0.621±0.029
First AC cosine	**0.749±0.037**	**0.709±0.029**	0.800±0.021
DC–AC score readout	**0.749±0.033**	0.702±0.023	**0.804±0.022**
GRP-128 first AC cosine	0.747±0.026	0.701±0.017	0.797±0.015

F.2 BENCH-A SUBSET DETAILS FOR PAIR HOLDOUT

Tables 14 and 15 summarize the nine parent groups and the default pair holdout protocol. Table 16 lists all 116 pairs with their original identifiers. The 58 related pairs link every derivative to its designated parent, covering all 67 models and providing the complete model roster. The 58 unrelated pairs connect models from different parent groups.

Table 14: **Parent groups in the pair holdout evaluation.** Group size includes the designated parent checkpoint and its derivatives. Group membership follows the recorded Bench-A parent checkpoint.

Parent checkpoint	Group size	Derivatives	Related pairs
`Qwen/Qwen2.5-1.5B-Instruct`	10	9	9
`deepseek-ai/deepseek-coder-1.3b-base`	10	9	9
`google/gemma-2-2b`	8	7	7
`meta-llama/Llama-3.2-1B-Instruct`	8	7	7
`Qwen/Qwen2-1.5B`	7	6	6
`google/gemma-2b`	7	6	6
`TinyLlama/TinyLlama-1.1B-Chat-v1.0`	6	5	5
`meta-llama/Llama-3.2-3B-Instruct`	6	5	5
`microsoft/phi-2`	5	4	4
Total	**67**	**58**	**58**

Table 15: **Bench-A subset and protocol for pair holdout.** Counts describe the default static evaluation used in the main paper.

Statistic	Value	Description
Models in evaluation	67	All appear in the complete pair list.
Parent groups	9	Each contains its designated parent checkpoint.
Related pairs	58	One parent paired with each derivative.
Unrelated pairs	58	Models from different parent groups.
Total pairs	116	The fixed pool for all pair holdout splits.
Class ratio	1:1	Balanced in both training and test sets.
Pair holdout splits	20	Fixed stratified splits with seed 42.
Training / test pairs	92 / 24	46 / 12 pairs of each class.
Prompts per model	600	The same aligned prompts for all models.
Maximum response length	128 tokens	The cached generation budget.

Table 16: **Complete model and pair roster for pair holdout on Bench-A.** All 116 pairs and 67 models are included. Related pairs appear first, with the parent checkpoint in Model A and its derivative in Model B, followed by unrelated pairs. Pair identifiers and model names match the experiment manifest.

Pair ID	Model A	Model B
Related pairs (58)		
pair_00001	meta-llama/Llama-3.2-3B-Instruct	iqwiki-kor/Llama3.2-3B-MP-RM
pair_00004	meta-llama/Llama-3.2-1B-Instruct	wy2001/creativestorywriter
pair_00005	Qwen/Qwen2-1.5B	LenguajeNaturalAI/leniachat-qwen2-1.5B-v0
pair_00007	TinyLlama/TinyLlama-1.1B-Chat-v1.0	NicholasCorrado/tinyllama-1.1b-chat-v1.0-ui-math-coding-dpo-2
pair_00008	TinyLlama/TinyLlama-1.1B-Chat-v1.0	not-lain/Finetuned_TinyLlama
pair_00010	Qwen/Qwen2.5-1.5B-Instruct	unsloth/Qwen2.5-1.5B-Instruct
pair_00011	meta-llama/Llama-3.2-3B-Instruct	jkazdan/llama3b-real-and-synthetic-sftsd0
pair_00012	Qwen/Qwen2.5-1.5B-Instruct	winstcha/Qwen2.5-1.5B-FT-v2
pair_00013	Qwen/Qwen2-1.5B	rombodawg/rombos_Replete-Coder-Qwen2-1.5b
pair_00015	meta-llama/Llama-3.2-1B-Instruct	phamhai/Llama-3.2-1B-CyberFrog
pair_00017	meta-llama/Llama-3.2-1B-Instruct	Isotonic/OrcaAgent-llama3.2-1b
pair_00019	meta-llama/Llama-3.2-1B-Instruct	Hjgugugjhuhjggg/mergekit-passthrough-owrmdht
pair_00022	TinyLlama/TinyLlama-1.1B-Chat-v1.0	nroggendorff/vegetarian-mayo
pair_00024	Qwen/Qwen2.5-1.5B-Instruct	Kukedlc/Qwen2.5-1.5B-Spanish-1.0-DPO
pair_00025	TinyLlama/TinyLlama-1.1B-Chat-v1.0	not-lain/mayo
pair_00030	Qwen/Qwen2-1.5B	macadeliccc/Samantha-Qwen2-1.5B
pair_00034	meta-llama/Llama-3.2-3B-Instruct	lunahr/thea-c-3b-25r
pair_00035	TinyLlama/TinyLlama-1.1B-Chat-v1.0	ismail-h/TinayLlama-1.1B-Scam-Conversation
pair_00039	Qwen/Qwen2.5-1.5B-Instruct	Syed-Hasan-8503/Qwen2.5-1.5B-Instruct-WO-Adam-mini
pair_00041	meta-llama/Llama-3.2-3B-Instruct	prithivMLmods/Llama-Doctor-3.2-3B-Instruct
pair_00043	Qwen/Qwen2.5-1.5B-Instruct	win10/Qwen2.5-2B-Instruct
pair_00044	Qwen/Qwen2-1.5B	Quardo/TR-Qwen2-1.5B-002-241130
pair_00047	Qwen/Qwen2.5-1.5B-Instruct	jaeyong2/Qwen2.5-1.5B-Instruct-Ja-SFT
pair_00048	Qwen/Qwen2-1.5B	oxyapi/oxy-1-micro
pair_00049	meta-llama/Llama-3.2-3B-Instruct	passing2961/Ultron-Summarizer-3B
pair_00050	Qwen/Qwen2.5-1.5B-Instruct	Vikhrmodels/Vikhr-Qwen-2.5-1.5B-Instruct
pair_00051	meta-llama/Llama-3.2-1B-Instruct	jeremierostan/llama32-1b-restorative
pair_00054	Qwen/Qwen2.5-1.5B-Instruct	sunjisen/Qweni
pair_00057	meta-llama/Llama-3.2-1B-Instruct	omrudra998/fifth
pair_00058	meta-llama/Llama-3.2-1B-Instruct	torchtorchkimtorch/Llama-3.2-Korean-GGACHI-1B-Instruct-v1
pair_00060	deepseek-ai/deepseek-coder-1.3b-base	woody72/albertus-deepseek-coder
pair_00061	microsoft/phi-2	nguyenthanhdo/phi-half
pair_00063	microsoft/phi-2	SuLLMerica/TeleQnA-Phi2-Phinetune
pair_00067	deepseek-ai/deepseek-coder-1.3b-base	stojchet/ktol0k1
pair_00069	deepseek-ai/deepseek-coder-1.3b-base	TokenBender/evolvedSeeker_1_3
pair_00076	deepseek-ai/deepseek-coder-1.3b-base	TechxGenus/CursorCore-DS-1.3B-LC
pair_00077	google/gemma-2-2b	MonsterMMORPG/fixed_sana
pair_00078	google/gemma-2b	mlabonne/Gemmalpaca-2B
pair_00080	google/gemma-2-2b	rinna/gemma-2-baku-2b
pair_00083	google/gemma-2b	vaishnavi18/zephyr-7b-gemma-sft
pair_00084	google/gemma-2b	kykim0/gemma-2b-ultrachat-sft
pair_00086	deepseek-ai/deepseek-coder-1.3b-base	TechxGenus/CursorCore-DS-1.3B
pair_00088	google/gemma-2b	pmking27/PrathameshLLM-2B
pair_00089	google/gemma-2-2b	shellzero/gemma2-2b-ft-law-data-tag-generation
pair_00090	Qwen/Qwen2.5-1.5B-Instruct	BAAI/Aquila-VL-2B-llava-qwen
pair_00092	google/gemma-2-2b	hishab/titulm-gemma-2-2b-v1.0
pair_00093	deepseek-ai/deepseek-coder-1.3b-base	TechxGenus/CursorCore-DS-1.3B-SR
pair_00095	microsoft/phi-2	thucdangvan020999/phi2_2.2B_mergkit_prunme
pair_00096	google/gemma-2-2b	rudrashah/RLM-spell-checker
pair_00097	deepseek-ai/deepseek-coder-1.3b-base	JetBrains/deepseek-coder-1.3B-kexer
pair_00101	microsoft/phi-2	rasyosef/phi-2-instruct-v0.1
pair_00104	google/gemma-2b	KartikGPT/Gemma-2b-finetune
pair_00105	google/gemma-2b	NexaAIDev/Octopus-v2
pair_00108	google/gemma-2-2b	jkazdan/gemma-2-2b_hs2_iterl_sftsd1
pair_00109	deepseek-ai/deepseek-coder-1.3b-base	stojchet/jd5
pair_00110	deepseek-ai/deepseek-coder-1.3b-base	stojchet/nl-all2
pair_00112	Qwen/Qwen2-1.5B	FreedomIntelligence/Apollo2-1.5B
pair_00113	google/gemma-2-2b	harishnair04/Gemma-medtr-2b-sft-v2
Unrelated pairs (58)		
pair_00000	Syed-Hasan-8503/Qwen2.5-1.5B-Instruct-WO-Adam-mini	not-lain/Finetuned_TinyLlama
pair_00002	not-lain/mayo	Isotonic/OrcaAgent-llama3.2-1b
pair_00003	nroggendorff/vegetarian-mayo	prithivMLmods/Llama-Doctor-3.2-3B-Instruct
pair_00006	macadeliccc/Samantha-Qwen2-1.5B	unsloth/Qwen2.5-1.5B-Instruct
pair_00009	macadeliccc/Samantha-Qwen2-1.5B	Syed-Hasan-8503/Qwen2.5-1.5B-Instruct-WO-Adam-mini
pair_00014	rombodawg/rombos_Replete-Coder-Qwen2-1.5b	prithivMLmods/Llama-Doctor-3.2-3B-Instruct
pair_00016	LenguajeNaturalAI/leniachat-qwen2-1.5B-v0	meta-llama/Llama-3.2-1B-Instruct
pair_00018	ismail-h/TinayLlama-1.1B-Scam-Conversation	meta-llama/Llama-3.2-1B-Instruct
pair_00020	winstcha/Qwen2.5-1.5B-FT-v2	not-lain/mayo
pair_00021	win10/Qwen2.5-2B-Instruct	omrudra998/fifth
pair_00023	macadeliccc/Samantha-Qwen2-1.5B	torchtorchkimtorch/Llama-3.2-Korean-GGACHI-1B-Instruct-v1
pair_00026	sunjisen/Qweni	TinyLlama/TinyLlama-1.1B-Chat-v1.0
pair_00027	Qwen/Qwen2.5-1.5B-Instruct	jkazdan/llama3b-real-and-synthetic-sftsd0
pair_00028	rombodawg/rombos_Replete-Coder-Qwen2-1.5b	jaeyong2/Qwen2.5-1.5B-Instruct-Ja-SFT
pair_00029	Qwen/Qwen2-1.5B	Vikhrmodels/Vikhr-Qwen-2.5-1.5B-Instruct
pair_00031	Vikhrmodels/Vikhr-Qwen-2.5-1.5B-Instruct	NicholasCorrado/tinyllama-1.1b-chat-v1.0-ui-math-coding-dpo-2
pair_00032	not-lain/Finetuned_TinyLlama	Hjgugugjhuhjggg/mergekit-passthrough-owrmdht
pair_00033	oxyapi/oxy-1-micro	passing2961/Ultron-Summarizer-3B
pair_00036	unsloth/Qwen2.5-1.5B-Instruct	meta-llama/Llama-3.2-1B-Instruct
pair_00037	Qwen/Qwen2-1.5B	sunjisen/Qweni
pair_00038	macadeliccc/Samantha-Qwen2-1.5B	omrudra998/fifth
pair_00040	jaeyong2/Qwen2.5-1.5B-Instruct-Ja-SFT	nroggendorff/vegetarian-mayo
pair_00042	Hjgugugjhuhjggg/mergekit-passthrough-owrmdht	iqwiki-kor/Llama3.2-3B-MP-RM
pair_00045	jaeyong2/Qwen2.5-1.5B-Instruct-Ja-SFT	Hjgugugjhuhjggg/mergekit-passthrough-owrmdht
pair_00046	ismail-h/TinayLlama-1.1B-Scam-Conversation	torchtorchkimtorch/Llama-3.2-Korean-GGACHI-1B-Instruct-v1
pair_00052	Vikhrmodels/Vikhr-Qwen-2.5-1.5B-Instruct	Hjgugugjhuhjggg/mergekit-passthrough-owrmdht
pair_00053	jaeyong2/Qwen2.5-1.5B-Instruct-Ja-SFT	lunahr/thea-c-3b-25r
pair_00055	LenguajeNaturalAI/leniachat-qwen2-1.5B-v0	sunjisen/Qweni
pair_00056	Isotonic/OrcaAgent-llama3.2-1b	prithivMLmods/Llama-Doctor-3.2-3B-Instruct
pair_00059	jaeyong2/Qwen2.5-1.5B-Instruct-Ja-SFT	not-lain/Finetuned_TinyLlama
pair_00062	hishab/titulm-gemma-2-2b-v1.0	rasyosef/phi-2-instruct-v0.1
pair_00064	FreedomIntelligence/Apollo2-1.5B	TechxGenus/CursorCore-DS-1.3B
pair_00065	TechxGenus/CursorCore-DS-1.3B	google/gemma-2-2b
pair_00066	nguyenthanhdo/phi-half	vaishnavi18/zephyr-7b-gemma-sft
pair_00068	unsloth/Qwen2.5-1.5B-Instruct	vaishnavi18/zephyr-7b-gemma-sft
pair_00070	jkazdan/llama3b-real-and-synthetic-sftsd0	mlabonne/Gemmalpaca-2B
pair_00071	TinyLlama/TinyLlama-1.1B-Chat-v1.0	rinna/gemma-2-baku-2b
pair_00072	oxyapi/oxy-1-micro	stojchet/nl-all2
pair_00073	lunahr/thea-c-3b-25r	rinna/gemma-2-baku-2b
pair_00074	mlabonne/Gemmalpaca-2B	winstcha/Qwen2.5-1.5B-FT-v2
pair_00075	google/gemma-2-2b	ismail-h/TinayLlama-1.1B-Scam-Conversation
pair_00079	jkazdan/llama3b-real-and-synthetic-sftsd0	shellzero/gemma2-2b-ft-law-data-tag-generation
pair_00081	not-lain/Finetuned_TinyLlama	stojchet/nl-all2
pair_00082	NexaAIDev/Octopus-v2	woody72/albertus-deepseek-coder
pair_00085	kykim0/gemma-2b-ultrachat-sft	macadeliccc/Samantha-Qwen2-1.5B
pair_00087	MonsterMMORPG/fixed_sana	google/gemma-2b
pair_00091	rasyosef/phi-2-instruct-v0.1	win10/Qwen2.5-2B-Instruct
pair_00094	TinyLlama/TinyLlama-1.1B-Chat-v1.0	pmking27/PrathameshLLM-2B
pair_00098	Hjgugugjhuhjggg/mergekit-passthrough-owrmdht	nguyenthanhdo/phi-half
pair_00099	SuLLMerica/TeleQnA-Phi2-Phinetune	Syed-Hasan-8503/Qwen2.5-1.5B-Instruct-WO-Adam-mini
pair_00100	TechxGenus/CursorCore-DS-1.3B	nguyenthanhdo/phi-half
pair_00102	MonsterMMORPG/fixed_sana	unsloth/Qwen2.5-1.5B-Instruct
pair_00103	rudrashah/RLM-spell-checker	wy2001/creativestorywriter
pair_00106	SuLLMerica/TeleQnA-Phi2-Phinetune	wy2001/creativestorywriter
pair_00107	stojchet/nl-all2	torchtorchkimtorch/Llama-3.2-Korean-GGACHI-1B-Instruct-v1
pair_00111	JetBrains/deepseek-coder-1.3B-kexer	phamhai/Llama-3.2-1B-CyberFrog
pair_00114	google/gemma-2-2b	unsloth/Qwen2.5-1.5B-Instruct
pair_00115	Vikhrmodels/Vikhr-Qwen-2.5-1.5B-Instruct	rudrashah/RLM-spell-checker

G Representation Ablations and Diagnostics

G.1 DC and First AC Components

Figure 9 gives the complete Agent500 component ablation. DC carries most of the dynamic source signal for every proxy and budget. First AC alone is consistently weaker, especially from one response, yet it remains far above chance and approaches DC as evidence accumulates. The joint DC–AC fingerprint either matches DC or changes it by at most a small amount. Its consistency across eight readers extends the central dynamic pattern beyond the four-model main roster.

The static relationship task reverses the component ordering. Table 13 isolates the same coordinates before the readout over two cosine scores. First AC improves over DC by 20.8 accuracy points and 17.9 AUC points. Combining the separately computed DC and AC cosine scores retains AC's accuracy and adds 0.4 AUC point. Thus the joint fingerprint is a common measurement space. The readout for each task determines which coordinate matters. Global trajectory location dominates dynamic identity, whereas coarse zero-mean evolution dominates static pair relationships.

Protocol for the main comparison. The eight representations in Table 4 share fixed proxy settings within each task. In the order Llama, Gemma, Ministral, and Qwen, attribution uses layers $2, 5, 17, 18$, and the pair task uses layers $12, 5, 16, 18$. All use prompt–response input and the same response token span under the 1024-token input limit. Each representation receives its own standardizer and source probe, fitted with the same 40-step Adam recipe on the original five Agent500 prompt folds. For Math100, each frozen fold model evaluates all responses. Metrics average the three grouping seeds within each fold and then the five fold models. For the pair task, each representation block contributes one cosine score, averaged over the 600 aligned prompts. The resulting scores use the same linear SVM with $C = 1$, training-split standardization, and twenty fixed splits for pair holdout.

G.2 Matched Temporal Representation Controls

We extend the Agent500 comparison with temporal encodings that share one frozen proxy pass and the same effective response tokens. This pass produces DCT modes zero through seven, the final token state, and the coordinate maximum. Mode zero is the all-token arithmetic mean. We compare individual blocks, pairwise pooling concatenations, and DCT bandwidths of two, four, and eight modes. Table 17 reports their Agent500 results.

Matched extraction and fitting. All variants use the fixed attribution layers specified above and the same prompt–response input. Each representation receives its own coordinate standardizer and source probe fitted on the Agent500 training prompts. The five prompt folds, 40-step Adam recipe, query budgets, grouping seeds, and evidence accumulator are shared across representations.

Learned temporal mixtures. Two additional controls jointly fit one or two global mixtures of the first eight modes with the Agent500 source probe. Let $\mathbf{z}_0, \ldots, \mathbf{z}_7$ denote mode blocks standardized using the training fold. Head h learns $\mathbf{a}_h \in \mathbb{R}^8$ and outputs

$$\widetilde{\mathbf{z}}_h = \sum_{m=0}^{7} \frac{a_{h,m}}{\|\mathbf{a}_h\|_2} \mathbf{z}_m. \tag{14}$$

One head returns a d-dimensional block, and two heads concatenate their outputs into a $2d$-dimensional fingerprint. These weights are shared across responses and are fitted jointly with the source classifier using only the outer-training prompts.

On Agent500, DC retains most source evidence, and its concatenation with first AC remains close to the exact mean at both endpoints. Increasing the bandwidth from $q = 2$ to $q = 4$ or $q = 8$ reduces accuracy at $K = 1$ by 2.8 and 7.0 points, respectively. Learning global mode mixtures changes accuracy by at most 0.3 percentage points relative to DC–AC at either endpoint. These comparisons isolate temporal encoding under the same attribution training protocol.

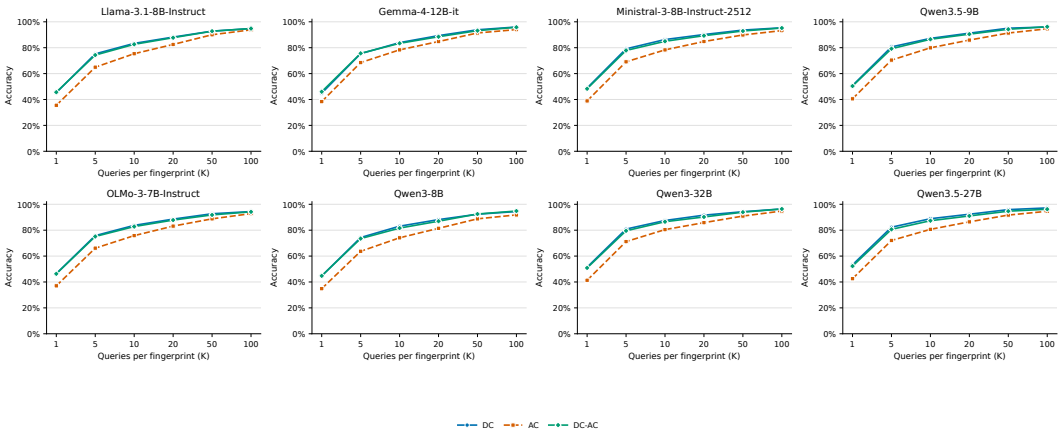


Figure 9: **Spectral-component accuracy on Agent500.** DC, first AC, and their unprojected concatenation use independently fitted probes at the same fold-local selected proxy layer.

Table 17: **Strictly matched temporal representations on Agent500.** Mean±standard deviation across four proxy means, after averaging three grouping seeds within each fold and then the five folds. Deviations measure proxy variation. All variants use identical proxy states, response tokens, fixed proxy layers, and the same source-probe protocol. d is the hidden dimension. "Learned" denotes additional temporal mixture weights. F1 is macro-F1.

Temporal representation	Dim.	Learned?	Acc.@1	F1@1	Acc.@100	F1@100
All-token mean (DC)	d	No	0.474±0.025	0.469±0.025	0.955±0.004	**0.942±0.006**
First AC only	d	No	0.383±0.018	0.379±0.018	0.939±0.006	0.922±0.007
Final token	d	No	0.159±0.045	0.157±0.046	0.659±0.089	0.603±0.099
Coordinate maximum	d	No	0.410±0.023	0.406±0.022	0.930±0.008	0.910±0.010
Mean + final	$2d$	No	0.386±0.026	0.386±0.026	0.896±0.023	0.874±0.027
Mean + maximum	$2d$	No	**0.478±0.015**	**0.476±0.015**	0.947±0.007	0.931±0.009
Final + maximum	$2d$	No	0.355±0.028	0.355±0.027	0.886±0.021	0.859±0.026
Learned mixture (1 head)	d	Yes	0.475±0.024	0.470±0.025	0.953±0.003	0.940±0.005
Learned mixture (2 heads)	$2d$	Yes	0.477±0.019	0.475±0.020	0.954±0.005	0.941±0.007
DC–AC (DCT $q = 2$)	$2d$	No	0.476±0.019	0.474±0.020	**0.956±0.005**	**0.942±0.008**
DCT $q = 4$	$4d$	No	0.448±0.013	0.448±0.013	0.947±0.005	0.932±0.006
DCT $q = 8$	$8d$	No	0.406±0.011	0.406±0.010	0.920±0.005	0.899±0.005

G.3 Evidence Accumulation

To isolate Stage 2, all accumulation rules reuse the same fitted DC–AC source heads, response scores, and query groups from the matched experiment. Probability averaging selects the class with the largest average normalized probability. Majority voting counts the classes predicted from individual responses and resolves ties by the fixed class order. BEA sums the log posteriors under the uniform enrollment prior. We evaluate all six query budgets with the same three grouping seeds in each outer fold.

Table 18 gives the results for each proxy. Every rule returns identical predictions at $K = 1$, and BEA has the highest mean accuracy for every proxy at each larger budget. As a tie sensitivity check, uniform random resolution of voting ties gives an expected four-proxy accuracy of 92.10% at $K = 100$, close to 92.30% under the fixed rule and below BEA's 95.55%.

G.4 Prompt Context at the Proxy Input

Figure 10 compares response-only input (r) with prompt–response input (ur). Both apply the DCT to response-aligned states. The curves nearly coincide across readers and budgets. Response-only input averages 0.5 point higher at $K = 1$, whereas prompt–response averages 0.1 point higher at $K = 100$. We report ur in the main tables because it retains the complete observed interaction and

Table 18: **Evidence accumulation by proxy on Agent500.** Accuracy (mean±standard deviation across five prompt folds, after averaging three grouping seeds per fold). All methods share the same response-level DC–AC predictions.

Accumulation	Acc.@1	Acc.@5	Acc.@10	Acc.@20	Acc.@50	Acc.@100
Llama-3.1-8B-Instruct						
Majority voting	0.456±0.011	0.622±0.012	0.727±0.012	0.807±0.010	0.879±0.011	0.918±0.015
Probability averaging	0.456±0.011	0.698±0.013	0.788±0.010	0.852±0.012	0.908±0.013	0.940±0.021
BEA	0.456±0.011	**0.743±0.011**	**0.826±0.013**	**0.877±0.012**	**0.927±0.016**	**0.948±0.013**
Gemma-4-12B-it						
Majority voting	0.460±0.012	0.625±0.016	0.734±0.011	0.807±0.018	0.875±0.021	0.916±0.031
Probability averaging	0.460±0.012	0.715±0.011	0.798±0.008	0.854±0.020	0.912±0.019	0.944±0.012
BEA	0.460±0.012	**0.756±0.006**	**0.834±0.008**	**0.885±0.014**	**0.932±0.014**	**0.958±0.015**
Ministral-3-8B-Instruct-2512						
Majority voting	0.483±0.010	0.654±0.008	0.757±0.009	0.831±0.005	0.892±0.010	0.926±0.012
Probability averaging	0.483±0.010	0.731±0.008	0.814±0.006	0.870±0.008	0.916±0.009	0.942±0.019
BEA	0.483±0.010	**0.779±0.007**	**0.849±0.009**	**0.893±0.007**	**0.929±0.011**	**0.954±0.017**
Qwen3.5-9B						
Majority voting	0.504±0.011	0.677±0.009	0.776±0.004	0.839±0.005	0.903±0.009	0.932±0.019
Probability averaging	0.504±0.011	0.747±0.006	0.828±0.004	0.874±0.004	0.926±0.004	0.944±0.014
BEA	0.504±0.011	**0.792±0.009**	**0.864±0.005**	**0.905±0.005**	**0.943±0.005**	**0.962±0.024**

Table 19: **Proxy input on Agent500.** Mean±standard deviation across four proxy means, after averaging the three grouping seeds. Deviations measure proxy variation. Both input views encode response-aligned states. F1 is macro-F1.

Proxy input	Acc.@1	F1@1	Acc.@100	F1@100
Response only (r)	**0.481±0.015**	**0.479±0.016**	0.954±0.007	0.953±0.007
Prompt–response (ur)	0.476±0.019	0.474±0.020	**0.955±0.005**	**0.954±0.006**

allows the proxy to interpret a response in its query context. The near tie shows that response content is sufficient for nearly all measured attribution performance.

H Geometry of Model Signatures

H.1 Construction and Enrichment in the Original Space

For each proxy, geometry uses the modal layer among its five Agent500 outer-fold selections. A modal tie is resolved by mean inner-validation accuracy across the five outer folds and then in favor of the shallower layer. For each source c, we average the response-level fingerprints over the 500 Agent500 prompts. We then normalize the DC and first AC blocks separately and concatenate them, matching the geometry displayed in the main-paper figure. Cosine distances are computed between the resulting 100 source centroids. The primary family-purity analysis covers the seven provider families represented by at least five models (87 models in total). Purity aggregation uses these sufficiently populated groups, while neighbor search retains all 100 models. Permutation tests randomly shuffle the seven family assignments while preserving group sizes.

Figure 11 summarizes purity across neighborhood sizes, and Table 20 reports the values for each proxy.

All four main proxies exceed the permutation null at Top-1 and Top-5 after Holm correction ($p \leq 8.0 \times 10^{-4}$ over eight tests). Their smallest Top-5 enrichment is $2.69\times$. The six pairwise correlations between their full 100×100 distance matrices range from 0.770 to 0.933, with median Spearman correlation 0.818. Their mean Top-10 neighborhood Jaccard similarities range from 0.459 to 0.618. Family structure is therefore present before dimensionality reduction and replicates across the four proxies.

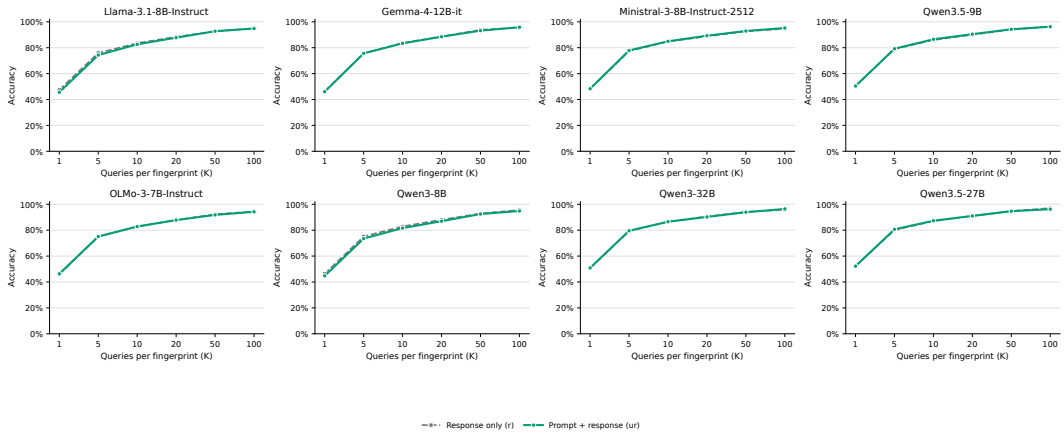


Figure 10: **Response-only and prompt–response proxy inputs.** Both variants encode states at response positions. Their proxy forward passes differ in the inclusion of the observed prompt.

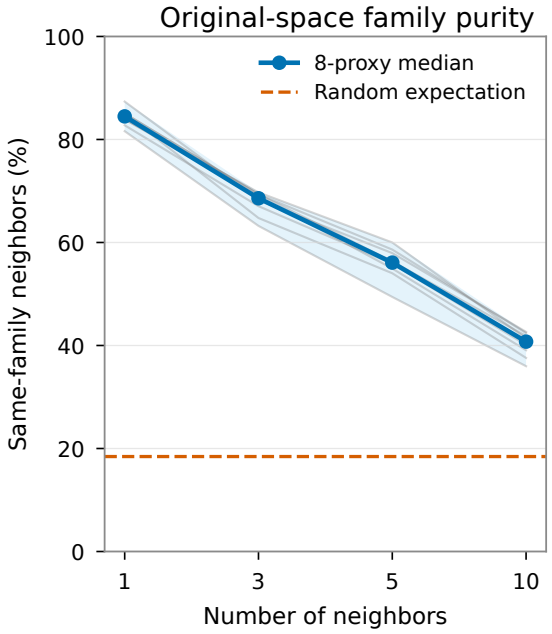


Figure 11: **Same-family neighbor purity before projection.** The highlighted curve shows median purity across eight proxy readers, and the dashed line marks the random expectation of 18.4%. Neighborhoods are measured in the original DC–AC cosine space.

H.2 Prompt and Projection Stability

To test sensitivity to the response panel, we partition the 500 prompts into five disjoint 100-prompt subsets and recompute every source centroid. Across the four main proxies, Spearman correlations between subset and full-panel distance matrices range from 0.940 to 0.994 (median 0.989). The model geometry is stable across disjoint prompt subsets.

For the two-dimensional visualization, we sweep five random seeds (0–4) and four perplexities (10, 15, 30, and 50). Across these 20 runs, trustworthiness@10 ranges from 0.957 to 0.973 with median 0.966. Every run retains more than twice the random family purity at Top-5. The displayed projection uses the separately fixed seed 42 and perplexity 15. Interpretation is restricted to local neighborhoods because t-SNE axes and global inter-cluster distances are arbitrary.

Table 20: **Same-family neighbor purity in the original DC–AC cosine space.** The random expectation is 18.4%. Values are proportions. The first four rows are the main proxy roster.

Proxy	Top-1	Top-3	Top-5	Top-10
Llama-3.1-8B	0.816	0.632	0.494	0.360
Gemma-4-12B	0.851	0.648	0.540	0.376
Ministral-3-8B	0.828	0.670	0.559	0.401
Qwen3.5-9B	0.851	0.693	0.563	0.418
OLMo-3-7B	0.851	0.693	0.586	0.424
Qwen3-8B	0.839	0.690	0.549	0.391
Qwen3-32B	0.839	0.697	0.600	0.414
Qwen3.5-27B	0.874	0.682	0.579	0.426

Table 21: **Cross-provider neighborhood stability across the four main proxies.** Mutual rank is the worse of the two directed ranks.

Model pair	Top-10	Med. rank
R1-Distill-Qwen-32B / QwQ-32B	4/4	5.0
Claude-Sonnet-4 / DeepSeek-V4-Pro	4/4	2.5
Gemma-4-31B / Gemini-3.1-Flash-Lite	4/4	3.5
MiniMax-M3 / Claude-Sonnet-4.6	2/4	11.0

H.3 Candidate Cross-Provider Neighborhoods

We evaluate four candidate neighborhoods first identified in the Qwen3.5-9B projection using a cross-proxy rank analysis. A pair passes when each model ranks the other within its Top-10 neighborhood for at least three of four main proxies and the median mutual rank is at most 10.

The distilled-Qwen pair is stable under all eight proxies. All three passing pairs indicate behavioral similarity in the observed responses. Establishing ancestry or shared training history requires independent evidence. MiniMax-M3 and Claude-Sonnet-4.6 are immediate neighbors under Qwen3.5-9B and several supplementary proxies, reaching the criterion for two of four main proxies. The cross-proxy threshold separates stable neighborhoods from salient single-proxy arrangements.

H.4 Interactive Three-Dimensional Atlas

Figure 12 shows the visualization interface developed to inspect the geometry of model signatures. It supports READER and LLM-DNA representations, multiple proxy or encoder configurations, and Agent500, Math100, cross-domain, and domain-centered views. Organization filters, model search, and neighborhood inspection expose structure that is difficult to read from a single static projection. The interface complements the paper figures with interactive exploration. Quantitative neighborhood analyses use the original signature space.

I Geometry of Model Signatures across Domains

We examine the geometry of model signatures under a substantial shift in query domain. The diagnostic uses the Qwen3.5-9B proxy at layer 18, its modal layer across the five Agent500 outer-fold selections, together with the response-only input view used for the analysis of model signatures. For each enrolled model endpoint, we average its DC–AC fingerprints over 500 Agent500 responses or 100 Math100 responses, normalize the two coefficient blocks separately, and concatenate them. Figures 13 and 14 fit t-SNE separately within each domain. Independent fits assign unrelated axes and global cluster positions. Both use cosine distance, seed 42, and perplexity 15.

I.1 Geometry Within Each Domain

The Math100 projection preserves substantial local organization despite the narrower domain. Recent GPT endpoints, Claude endpoints, several Qwen generations, and Llama endpoints form recog-



Figure 12: **Three-dimensional Model Provenance Atlas.** The screenshot shows Agent500 model signatures read by the Qwen3-32B READER proxy. The interface supports provenance-method, proxy, domain, organization, and model selection together with interactive neighborhood inspection.

Table 22: **Cross-domain matching in the original fingerprint space.** Each Agent500 centroid retrieves among all 100 Math100 centroids. "Wins other" is the probability that the matched model has higher cosine similarity than a randomly selected different model.

	Median cosine		Same-model retrieval			Wins
Space	**Same**	**Other**	**Top-1**	**Top-5**	**MRR**	**other**
Raw	0.884	0.847	0.090	0.430	0.245	0.851
Centered	0.416	−0.013	0.250	0.670	0.428	0.924

nizable neighborhoods in both panels. Their global arrangement changes, and several cross-family neighborhoods are rearranged. Local family structure thus persists within Math100. The joint projection and analyses in the original space below quantify cross-domain identity.

I.2 Joint Geometry and Shared Domain Shift

We next fit one t-SNE to all 200 centroids. Each line connects the Agent500 and Math100 centroid of the same endpoint. In the raw joint projection (Figure 15), circles and triangles separate into two broad regions. The query domain thus contributes a shared component large enough to dominate the two-dimensional neighborhood graph. t-SNE distorts global metric distances, so line lengths serve as visual guides.

To isolate residual structure, we perform one diagnostic transformation: subtract the mean of the 100 endpoint centroids within each domain, then renormalize every centered signature before the joint projection. This unsupervised centering removes a domain-wide offset and is used only for this geometry diagnostic. Figure 16 shows that many family-local neighborhoods align more closely after this operation, whereas exact endpoint pairs remain dispersed.

Table 22 evaluates the original 8,192-dimensional fingerprints. Before centering, the matching endpoint ranks first for 9 of 100 queries, yet it outranks 85.1% of different endpoints. Domain centering raises exact Top-1 matching to 25%, Top-5 matching to 67%, and the pairwise win rate to 92.4%.

The cosine values across the two rows have different baselines because centering changes the coordinate origin.

Together, the projections and retrieval metrics in the original space indicate a two-level geometry. A broad query-domain component shifts the ecosystem as a whole, while model- and family-specific residuals persist underneath it. This pattern is consistent with Math100's within-domain organization and its lower attribution accuracy without retraining.

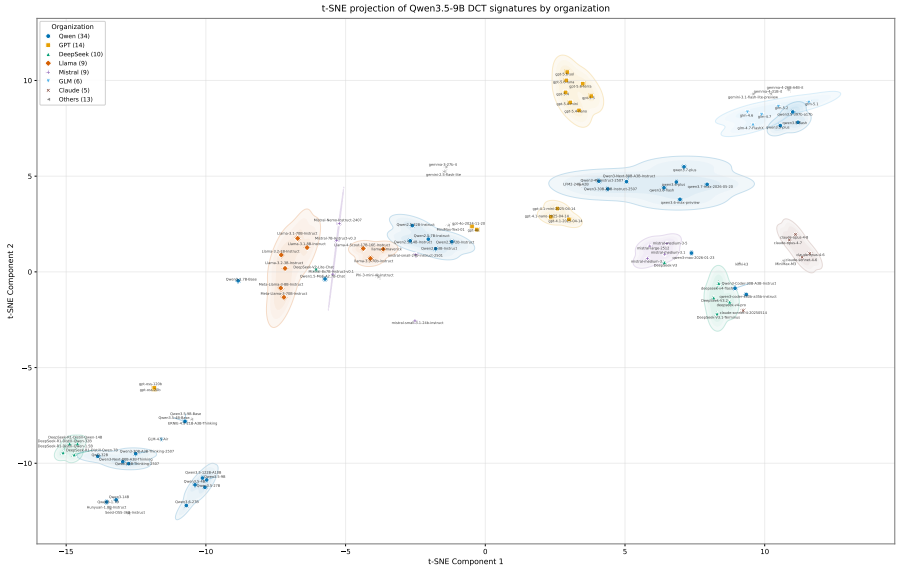


Figure 13: **Geometry of Agent500 model signatures.** The 100 endpoint centroids are computed from 500 query-varying interactions per endpoint. Contours visualize dense same-family regions. Quantitative results use the original fingerprint space.



Figure 14: **Geometry of Math100 model signatures.** The same construction is applied to 100 mathematical interactions per endpoint. This projection is fit independently from Figure 13.

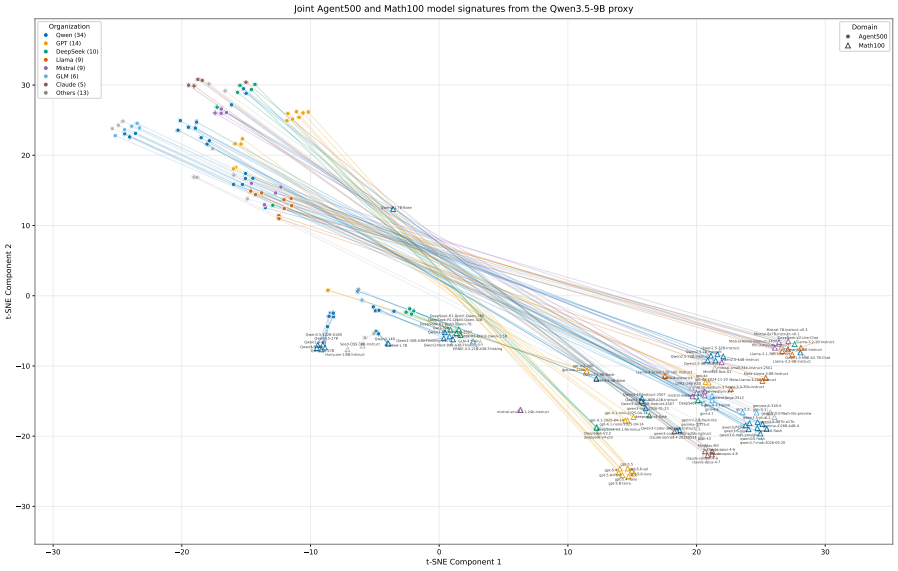


Figure 15: **Raw joint Agent500–Math100 projection.** Circles denote Agent500, triangles denote Math100, and lines connect the same model endpoint across domains. One t-SNE is fit to all 200 centroids.

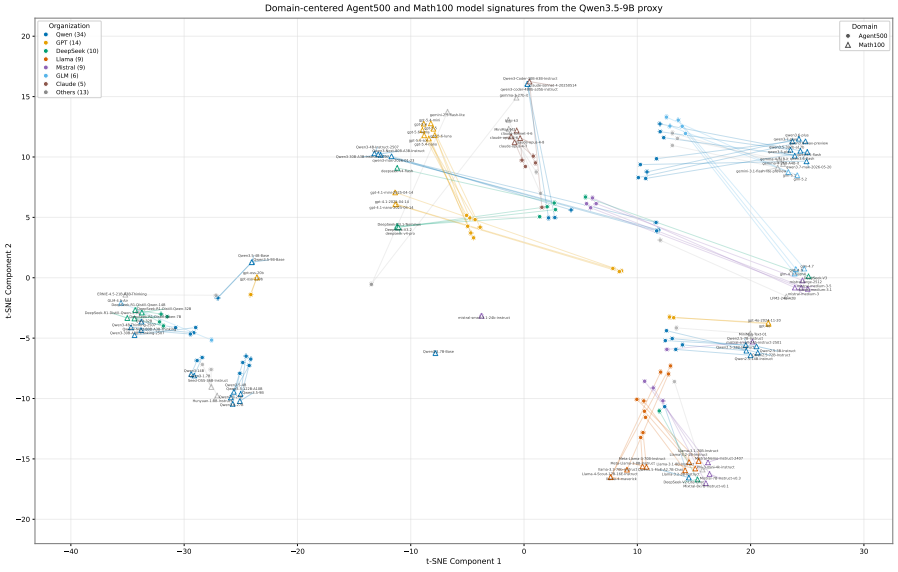


Figure 16: **Domain-centered joint projection.** Before fitting t-SNE, each domain mean is removed in the original fingerprint space and every endpoint signature is renormalized. Domain centering is used only for this geometry diagnostic.

J Reproducibility and Scope

J.1 Reproduction Checklist

Table 23 maps each reported result to the information needed to reproduce it. Released configurations record the proxy checkpoint, layer-selection setting, input view, DCT modes, split seed, grouping seeds, classifier settings, and evaluation protocol. Intermediate fingerprints are keyed by benchmark, proxy, layer, and input view so that representation extraction and statistical evaluation can be audited independently.

Table 23: **Reproduction map for the reported experiments.**

Result category	Split, fitting, and repetitions
Agent500	Five outer prompt folds (seed 42), four-fold inner selection over all blocks by response-level accuracy, fold-local standardizer/probe, and grouping seeds 42–44 at six budgets
Cross-task representations	Eight representations in Table 4, fixed layers for each task, frozen Math100 transfer, and twenty pair splits
Agent500 temporal controls	Fixed attribution layers, one shared proxy pass, the same five Agent500 prompt folds, and the same Adam recipe
Evidence accumulation	Identical DC–AC heads and query groups, three accumulation rules, five prompt folds, and three grouping seeds at six budgets
Length shift	Original prompt folds and frozen Agent500 estimators, with each response assigned to the fold model that excluded its prompt
Math100	Five frozen fold models evaluated independently on 100 new prompts, with metrics averaged across fold models
Bench-A pair holdout	Modal Agent500 layer, twenty fixed 92/24 pair splits, and a split-local standardizer and SVM
Bench-A transfer	Twenty splits for model holdout and all 36 splits for parent group holdout over nine parent groups, with balanced train/test pairs and a split-local standardizer and SVM
Geometry	Modal Agent500 layer, 500 prompts, neighbor metrics in the original space, stored prompt partitions, projection sweep, and permutations

Numerical determinism. The DCT coefficients and evidence grouping are deterministic once tokenizer outputs and seeds are fixed. Floating-point proxy inference and GPU optimization can introduce small implementation-level differences. We retain per-fold response log-posteriors in addition to aggregate tables, allowing all K-grouped metrics to be recomputed from cached scores.

API endpoints. Agent500 records the endpoint identifier and collection metadata for every API response. The released responses and metadata define a fixed benchmark snapshot as served endpoints evolve. Subsequent comparisons should preserve the versioned identifiers in Table 8.

J.2 Computational Profile

The primary experiments and runtime profiling use NVIDIA H200 accelerators with 140 GB HBM. The matched ablations in Section 4.5 use NVIDIA A100 accelerators with 80 GB HBM. Runtime measurements in Table 9 use a single H200 after warm-up.

Table 9 separates learned capacity from the parameters required by the audit forward path. READER fit time starts from cached candidate-layer fingerprints and covers all inner-fold probe fits, layer selection, and the final outer-training probe fit. Audit-time latency covers proxy forward passes. Other frozen-representation methods start from cached enrollment representations and include fold-local standardization. READER and the DNA adaptations optimize their 100-way linear heads for 40 full-batch Adam steps. LLMmap trains its $Q = 1$ head for six selected epochs. DeBERTa is fine-tuned end to end for two epochs. Trainable parameter counts exclude standardizer statistics. Each inference measurement covers tokenization, representation extraction, and the 100-way score after model loading, with disk I/O excluded. We report the median of three warmed runs over 256 interactions using each implementation's native batch size. LLMmap's shared E5 encoder is counted

once and covers both prompt and response encoding. The dagger denotes adaptations from a fixed-panel protocol to dynamic response-level attribution.

For READER, the inference count comprises the text embedding, Transformer blocks through the selected hidden state, and the 100-way linear head. The forward path exits at the selected block, before subsequent Transformer blocks, the language-model output head, and unused vision modules. The latency measurement uses the same early exit.

The frozen-proxy pass over response tokens dominates READER's cost and parallelizes across proxy, source, and prompt. Once fingerprints are cached, evidence regrouping and all query-budget evaluations reuse the same arrays. Target models contribute text responses. The frozen proxy performs audit-time feature extraction, and the source probe contains all trainable parameters. Storage scales as $O(Nd)$ for two coefficients per response, and the probe scales as $O(Cd)$. The unprojected design trades larger cached features for a deterministic, projection-free representation.

The table reports Agent500 because its unit is an individual dynamic interaction. MPT and PhyloLM operate on fixed-prompt model pairs in the static Bench-A evaluation, whose pair-construction costs and per-example latency use a different computational unit.

J.3 Intended Scope

READER targets dynamic auditing in an enrolled ecosystem through stateless interactions. Its DC–AC fingerprint also supports static relationship auditing within a shared measurement space. Dynamic identification uses the source probe and evidence accumulator. Static model pairs use a relationship readout over the same spectral blocks. The protocol assumes meaningful candidate identities, labeled enrollment responses, a fixed ecosystem, and independent requests. Unknown-source rejection, endpoint drift, conversational state, and adversarial evasion remain open deployment conditions. The tests without retraining expose sensitivity to response coverage and domain shift.

Attribution may expose sensitive information. Deployments should document collection policy, minimize stored text, restrict access to fingerprints and source scores, and provide manual review for high-stakes decisions.

K Limitations

READER assumes a fixed enrolled ecosystem, stateless requests, and source-independent prompt selection. It cannot reject unknown sources or model drift, and proxy inference adds cost. Tables 1 and 2 show substantial evidence loss under severe response shortening and domain shift. Static auditing requires aligned prompts and a separate readout. Open-set attribution, drift detection, and transfer across domains remain deployment challenges.

L Large Language Model Usage

We used Claude Opus 4.6/4.7 as coding assistance tools for implementation, debugging, and experiment-running support. We used GPT-5.5/GPT-6-Astra to assist with manuscript polishing and English proofreading. All technical claims, experimental results, tables, and final manuscript content were reviewed and approved by the authors.