BEYOND FLAT ATTRIBUTION: RECOVERING BOOLEAN EVIDENCE FORMULAS FOR BLACK-BOX LANGUAGE MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Post-hoc context attribution for closed LLMs asks which spans of a prompt support a claim the target asserts. Most black-box methods return flat per-span scores, which cannot express that either of two spans may satisfy the same evidence requirement. We formulate attribution as the recovery of a Boolean *evidence formula* $\varphi_c(S) = \bigwedge_r \bigvee_{v \in G_r} \mathbb{1}[v \in S]$, a monotone AND-of-OR representation for which flat span sets have a clause-averaged F1 ceiling of $\frac{1}{m}\sum_r 2/(1+w_r)$. LEAF builds a fixed set of candidate formulas from LLM-derived span roles and uses counterfactual queries to select among them. Under a registry-only answering instruction, we evaluate on synthetic Easy/Hard, two long-context extensions, and a curated 500-example fact-verification benchmark from HoVer, FEVER, and SciFact. Under the primary binary-feedback protocol, LEAF achieves Formula F1 of 0.83–0.99. Its margin over the strongest reported baseline ranges from $+0.10$ on Natural to $+0.49$ on the 100-span setting. The Hard result extends across four closed targets and Llama-3.3-70B. Held-out behaviour and weak-cartographer tests probe the roles of the metric and LLM prior.

1 INTRODUCTION

Post-hoc *context attribution* for closed LLMs asks which spans of a prompt support a generated claim (Cohen-Wang et al., 2024; Zhang et al., 2025; Chang et al., 2025; Pan et al., 2026; Wang et al., 2025). Most published black-box methods return flat per-span scores. Yet evidence can have dependencies: an eligibility judgment may require both a rule and a value recorded in either of two places; multi-hop questions may combine distinct facts while admitting alternative passages. A flat ranking cannot distinguish alternatives from jointly required spans. We study this distinction through masked-span queries to models instructed to answer only from a supplied registry.

A real example. Consider the HoVer (Jiang et al., 2020) claim *"Christopher Nolan, who directed 'Inception', was born in London"* over seven Wikipedia paragraphs: S_1 states that *Inception* was directed by Nolan (the bridge), S_2 and S_3 are two separate paragraphs that each independently state Nolan was born in London, and S_4–S_7 are distractor paragraphs about distribution and box-office figures (paragraph text in Appendix D). Under the registry-only evaluation prompt, the target needs S_1 and *either* S_2 or S_3; the annotated structure is $S_1 \wedge (S_2 \vee S_3)$. A flat output faces a dilemma. Returning all three spans, read as $S_1 \wedge S_2 \wedge S_3$, gives perfect Span F1 but asserts that deleting S_3 breaks the claim when it does not; under the clause-matched metric of §4 this scores 0.56. Returning $\{S_1, S_2\}$ scores 0.83, the ceiling we prove in §2, but now drops S_3 as irrelevant (Span F1 0.80). No flat span set scores higher on Formula F1; the formula $S_1 \wedge (S_2 \vee S_3)$ scores 1.00. Figure 1 (Appendix D) shows the same failure on a synthetic eligibility prompt.

We re-formulate context attribution as the recovery of a Boolean *evidence formula*

$$\varphi_c(S) \;=\; \bigwedge_{r=1}^{m} \bigvee_{v \in G_r} \mathbb{1}[v \in S]\,, \tag{1}$$

where S is the set of included context spans and G_r is the r-th *requirement group* of redundant alternatives. The class is monotone in S and expresses both multi-hop AND and redundant OR,

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.

unlike a flat span set (§2). LEAF (§3) constructs a fixed beam of formulas from LLM-derived span roles. Target-model ablations reweight those candidates under a Gaussian pseudo-likelihood, while posterior disagreement selects the next query (Houlsby et al., 2011).

Our focus is *structured evidence logic*: on the curated natural benchmark LEAF and the strongest baseline have the same mean Span F1 (0.86) while LEAF has higher Formula F1 (§5), and we compare synthetic gaps with the flat-output ceiling to separate the representational bound from the observed shortfall of flat baselines.

Contributions. (i) Applying AND-of-OR evidence formulas (Eq. 1) to black-box LLM context attribution, with a clause-averaged F1 ceiling for flat span sets (§2). (ii) LEAF, which uses a fixed, role-derived formula beam and target-model queries to select among its candidates (§3). (iii) A four-benchmark evaluation (synthetic Easy, Hard 25-span, long-context to 100 spans, and a 500-example natural benchmark from HoVer, FEVER, and SciFact with annotated disjunctive evidence; §4) on which LEAF leads six reported baselines by $+0.10$ to $+0.49$ in Formula F1 (§5). (iv) Tests across four closed and one open target, held-out behaviour prediction, a weak-cartographer ablation, and a formula-shape stress test that probe alternative explanations for the gains.

2 EVIDENCE FORMULAS

Setup. Let P have spans $C_1, \ldots, C_n$ and let c be a claim the target asserts on the full prompt P. The claim may be elicited from the target or supplied for verification. For $\mathbf{z} \in \{0,1\}^n$, $P(\mathbf{z})$ is P with C_i included iff $z_i = 1$, and $S(\mathbf{z}) = \{i : z_i = 1\}$. Let $y(\mathbf{z}) \in [0,1]$ measure how strongly $f(P(\mathbf{z}))$ asserts c via a deterministic judge.

Hypothesis class. An *evidence formula* $\varphi_c(S) = \bigwedge_{\ell=1}^{m} \bigvee_{v \in G_\ell} \mathbb{1}[v \in S]$ is a monotone CNF; we write $\mathcal{H}_{n,m,r}$ for the class with $\leq m$ clauses and $\leq r$ alternatives per clause. The class is monotone: $S \subseteq S' \Rightarrow \varphi_c(S) \leq \varphi_c(S')$. A flat span set $\hat{S}$ carries no Boolean structure. For evaluation, we read a flat set of any size as the AND-of-singletons $\hat{\varphi}(S) = \bigwedge_{i \in \hat{S}} \mathbb{1}[i \in S]$, which lies in $\mathcal{H}_{n,m,1}$ whenever $|\hat{S}| \leq m$. This convention distinguishes a flat output from a method that explicitly predicts OR clauses.

Proposition (F1 ceiling for flat outputs). Let $\varphi^\star = \bigwedge_{\ell=1}^{m} C_\ell^\star$ with $|C_\ell^\star| = w_\ell$. For any flat $\hat{S}$, the bipartite-matched mean-per-clause F1 (§4) satisfies

$$F1(\hat{S}, \varphi^\star) \;\leq\; \frac{1}{m} \sum_{\ell=1}^{m} \frac{2}{1 + w_\ell}. \tag{2}$$

Proof. A predicted singleton matched to $C_\ell^\star$ contributes at most $2/(w_\ell + 1)$, with equality only when that span belongs to the clause. Each gold clause receives at most one match, and the score is divided by $\max(|\hat{S}|, m) \geq m$. Equality in the bound therefore requires all m positive terms and a denominator of m. □ Equality thus requires $|\hat{S}| = m$ and a one-to-one matching from the selected spans to the gold clauses such that each matched span belongs to its gold clause. This is stronger than selecting one member of each clause when clauses overlap: the same span cannot stand for two predicted singleton clauses.

On Easy Redundant ($m = 2$, $w = (1, 2)$) the ceiling is $(1 + 2/3)/2 \approx 0.83$; the flat BM25 baseline reaches 0.83 while LEAF reaches 0.99 (§5). Direct LLM can emit OR clauses and is not subject to this ceiling. The bound is a property of the specified flat-output interpretation and metric, not a limit on all attribution methods.

What the ceiling does and does not say. The numerical bound in Eq. 2 is specific to clause-matched Formula F1; Span F1 has no analogous bound. A flat output cannot itself distinguish jointly necessary spans from alternatives, but a later structural decoder could add that information. We therefore report complementary metrics (§4): Formula F1 compares annotated clause structure, Span F1 compares selected spans, and held-out behaviour accuracy compares predictions with target responses on sampled ablations. None alone establishes that an annotated formula is uniquely determined by the target's behaviour.

Algorithm 1 LEAF

Require: Prompt P; target f; planner g; claim c; budget B

PRI $\leftarrow$ CARTOGRAPH(g, P, c) ▷ one planner call: role label + prior per span
BEAM $\leftarrow$ INITBEAM(PRI; K) ▷ K fixed candidates, incl. the OR-anchor
$\mathcal{O} \leftarrow \{\text{QUERY}(P), \text{QUERY}(P_\emptyset)\}$
for $s \in$ top-2 by prior **do**
  $\mathcal{O}$ += QUERY$(P \setminus \{s\})$
end for
BEAM.REWEIGHT$(\mathcal{O})$
while $|\mathcal{O}| < B$ **do**
  $Q_t \leftarrow$ CANDIDATES(BEAM, P) ▷ unqueried masks from the grammar, $O(Kmr+n)$; stop if empty
  $\mathbf{z}^\star \leftarrow \arg\max_{\mathbf{z} \in Q_t} A(\mathbf{z}; \text{BEAM})$ ▷ posterior disagreement $p(1-p)$
  $\mathcal{O}$ += QUERY$(P(\mathbf{z}^\star))$; BEAM.REWEIGHT$(\mathcal{O})$
end while
return BEAM.MAP(τ) ▷ fewest clauses within τ of the best

3 THE LEAF ALGORITHM

LEAF (Algorithm 1) pairs an LLM-based *proposer* with a black-box target *validator*. A single *cartographer* planner call labels each span with one of five structural roles—RULE, DATA, RED._DATA, DISTRACTOR, or IRRELEVANT—and a prior in $[0, 1]$; the vocabulary is structural rather than domain-specific. A weak-roles ablation in §5 replaces it with three classes (SUPP., IRREL., UNKN.). A beam of $K = 16$ formula candidates is constructed once from these labels—clause and OR variants over the top-prior spans, including an explicit *OR-anchor* $[\text{RULE}_1] \wedge ([\text{DATA}_1] \vee [\text{RED._DATA}_1])$ priming disjunctive redundancy. The candidate set is *fixed* thereafter: observations reweight its K formulas and steer the queries, but cannot recover a formula absent from the initial beam. Each candidate is reweighted under a Gaussian pseudo-likelihood $\log L(\varphi \mid \mathcal{O}) = -\sum_q (y_q - \varphi(S_q))^2/(2\sigma_0^2)$ with $\sigma_0 = 0.15$; among formulas within $\tau = 0.5$ log-likelihood of the best, we return the one with fewest clauses (soft Occam factor). The next intervention is selected by posterior-disagreement: $\mathbf{z}^\star = \arg\max_{\mathbf{z}} p(1-p)$ over a candidate pool Q_t generated from beam-derived ablation patterns (Houlsby et al., 2011), where p is the current beam-weighted probability that the claim is preserved.

Intervention grammar. The candidate pool Q_t in Algorithm 1 is generated by enumerating four families of interventions: *keep-only* of each beam member's predicted spans; *remove-each-clause* (one clause at a time from each beam member); *remove-each-alternative* (one alternative from one clause); and *leave-one-out* over each individual span. The total pool size is $O(Kmr + n)$, where K is the beam size, m the maximum number of clauses, r the maximum clause width, and n the number of spans. Previously queried masks are excluded from Q_t. Every query is a masked copy of the original prompt, so the validator never needs log-probabilities, embeddings, or any access beyond the text API.

Cost. At most B target calls plus one cartographer call, with content-hash caching across all queries. We use $T = 0$, but API responses are not guaranteed to be deterministic. Hyperparameters, prompt descriptions, and the judge are in Appendix G; a token-level cost comparison is in §5 and Appendix H.

Search scope. The unrestricted hypothesis class can be large: for fixed m, r, it contains at most $O(n^{mr})$ distinct clause lists. LEAF does not enumerate this class. The cartographer creates a fixed beam of $K = 16$ candidates, and target queries reweight those candidates (Algorithm 1). If the desired formula is absent from the beam, query selection cannot recover it exactly. The observed budget at which performance saturates is therefore an empirical result, not a consequence of the class-size calculation. Appendix F discusses this scope and the distinction between a formula's behaviour and its annotated syntax.

4 BENCHMARKS AND SETUP

We use four benchmarks spanning synthetic–natural and short–long-context axes. Hyperparameters are in Appendix G; natural-benchmark curation, annotation protocol, and shortened excerpts for the worked example of §1 are in Appendix D.

Synthetic Easy (200 **ex,** 4**–**9 **spans).** Four phenomena, 50 examples each: T1 single-hop, T2 multi-hop, T3 redundant ($[v_{\text{rule}}] \wedge ([v_{\text{data}}] \vee [v_{\text{red}}])$), T4 fine-grained.

Synthetic Hard (100 **ex,** 25 **spans).** The redundant construction hardened with semantically-close distractors, paraphrased redundancy without role-label leakage, 3 competing rule-shaped spans, and long mixed paragraphs.

Long-context scaling (200 **ex).** The Hard construction extended to 50 and 100 spans (100 examples each), isolating context length.

Natural (500 **ex from three sources).** Curated from HoVer multi-hop fact verification (Jiang et al., 2020), FEVER claim-evidence pairs (Thorne et al., 2018), and SciFact scientific-claim verification (Wadden et al., 2020), in proportions $300/100/100$. Source-by-source numbers are broken out in Table 10 (Appendix D). From each source we start with a supplied verification claim, rather than an unconstrained model-generated statement, and retain examples where two or more supporting paragraphs independently entail the same claim fact (disjunctive redundancy) and a separate paragraph supplies the bridge/context needed to identify the relevant entity. Candidates are verified against the filtering target with seven masked ablations and jointly annotated by three internal annotators who did not author the method, were blind to model outputs, and followed a written guideline (pairwise Cohen's $\kappa = 0.81$; 23% retention from $\sim$ 2,200 candidates; Appendix D). Gold: $\text{BRIDGE} \wedge (\text{EVIDENCE}_1 \vee \text{EVIDENCE}_2)$ with $r \geq 2$ alternatives on a small subset. Gold structure on natural text is defined operationally relative to the filtering target; it is not a target-independent annotation of which evidence a model must use. The cross-filtering-target check in §5.2 probes sensitivity to that choice. The AND-of-OR class covers both "$\text{RULE} \wedge (\text{DATA} \vee \text{DATA})$" in the synthetic setting and "$\text{BRIDGE} \wedge (\text{EVIDENCE} \vee \text{EVIDENCE})$" in the natural setting with no algorithmic change.

Targets. GPT-4o-mini (primary, all ablations), GPT-4o, Claude Sonnet 4.6, Gemini 2.5 Flash (model identifiers recorded at evaluation time; results cached), and open Llama-3.3-70B-Instruct, all at $T = 0$ with content-hash caching. The planner is GPT-4o-mini across all targets. A strict system instruction requires the target to reply "Cannot determine from the registry" on missing facts; the judge scores any non-assertion as $y = 0$. This setup tests claim preservation under the registry instruction, rather than unconstrained generation.

Baselines and metrics. Six baselines: **BM25** (lexical); **Direct LLM** (one answer-conditioned call to the same planner model, prompted to return an AND-of-OR formula and free to emit OR clauses; it is a one-shot structured proposer with no validation, and is model-agnostic); **LOO**; **Random sparse** (B random masks fit with Lasso, the ContextCite design (Cohen-Wang et al., 2024)); **Flat MAB** (each span is an arm; UCB acquisition over B target queries; output top-k spans converted to AND-of-singletons — a flat-output analogue of Pan et al. (2026)'s combinatorial bandit, not a re-implementation, as their reward uses target log-probabilities that two of our closed targets do not expose); **TracLLM** (Wang et al., 2025) (run at our budget range; its native regime is $\geq$ 10K-token contexts and hundreds of queries). Budgets $B \in \{4, 8, 16, 32\}$. Flat rankings become sets by keeping spans scoring at least half the example's maximum; LOO keeps spans whose removal breaks the claim. **Formula F1** bipartite-matches predicted clauses to gold by IoU; the per-clause F1 of each matched pair is summed, and the sum is divided by $\max(\hat{m}, m)$ where $\hat{m}, m$ are the number of predicted and gold clauses, so that any unmatched predicted or gold clause contributes $\text{F1} = 0$ to the average. This metric compares formula structure directly. **Span F1** is per-span set agreement against the union of gold clauses. **Held-out behaviour accuracy** (§5.4) tests whether the predicted formula correctly predicts preservation on $H = 16$ subsets sampled per example; it scores behavioural agreement rather than syntactic similarity to the gold formula.

	Easy B=8	**Hard** B=16	**Long-50** B=16	**Long-100** B=16	**Natural** B=16	**Multi-model** avg. Hard B=16
BM25	0.75	0.50	0.36	0.26	0.38	0.50
Direct LLM	0.67	0.54	0.40	0.27	*0.74*	0.54
Leave-one-out	*0.87*	0.41	0.30	0.18	0.52	0.39
Random sparse	0.77	0.57	0.43	0.31	0.55	0.54
Flat MAB (ours)	0.80	*0.59*	*0.46*	*0.34*	0.58	*0.55*
TracLLM	0.72	0.33	0.25	0.17	0.49	0.32
LEAF (ours)	**0.99**	**0.96**	**0.91**	**0.83**	**0.84**	**0.93**
Gap	+0.12	+0.37	+0.45	+0.49	+0.10	+0.37
95% CI	[+.08, +.16]	[+.30, +.44]	[+.38, +.53]	[+.40, +.57]	[+.06, +.14]	–

Table 1: **Headline Formula F1 under binary target feedback.** The first five columns show LEAF at the displayed budget and the corresponding baseline scores; the last shows mean performance across five Hard targets at $B = 16$ (Table 5). Target-querying baselines in this table use the binary judge. Italics mark the best baseline in each column. The multi-model comparator is the best *single method averaged across targets*, not a different winner selected for each target. Gaps use unrounded scores before the displayed two-decimal truncation. Paired bootstrap 95% CIs (1000 resamples over examples) are reported for the first five gaps; no CI is available for the multi-model average. LEAF CIs: Easy $[0.98, 1.00]$, Hard $[0.92, 0.99]$, Long-50 $[0.85, 0.96]$, Long-100 $[0.75, 0.90]$, Natural $[0.81, 0.87]$.

5 RESULTS

Headline. Table 1 reports Formula F1 at the displayed budget for each benchmark, alongside query-based baselines at the same target-query budget and one-shot baselines for reference. LEAF has the highest mean score in every setting. Across the five benchmark columns with paired bootstrap intervals, its margins range from $+0.10$ on Natural to $+0.49$ on Long-100; all five intervals exclude zero. The multi-model average also favors LEAF, but we do not report an interval for that aggregate. Per-budget saturation is discussed in Appendix C.

5.1 SYNTHETIC AND LONG-CONTEXT

On Redundant, the strongest flat baseline reaches the proposition's $(1 + 2/3)/2 \approx 0.83$ ceiling (BM25; per-phenomenon results in Table 8, Appendix A); LEAF reaches 0.99. On Single-hop, Multi-hop, and Fine-grained at least one flat baseline also reaches $F1 = 1.00$, but no single flat baseline is top on all four phenomena, and every flat baseline is bounded by the ceiling on Redundant. On Hard (25 spans, paraphrased redundancy) LEAF attains 0.96 vs. best-baseline 0.59 at the same budget. The observed margin grows with context length ($+0.37 \rightarrow +0.45 \rightarrow +0.49$ at $25, 50, 100$ spans).

Where the observed gap lies. Table 2 gives an arithmetic decomposition of each synthetic gap around the flat-output ceiling. The gold formulas on Hard and long-context share the Redundant template ($m = 2$, $w = (1, 2)$), so this ceiling is 0.83 throughout. We call $0.83 -$ best flat the *flat headroom*, and LEAF $- 0.83$ the *above-ceiling margin*. On Easy-Redundant, BM25 reaches the ceiling and LEAF exceeds it by 0.16. On Hard, the best flat baseline scores 0.59: 0.24 of the observed 0.37 gap is flat headroom, and 0.13 is above the ceiling. On Long-100, LEAF reaches 0.83 while the best flat baseline scores 0.34, leaving 0.49 of flat headroom. These are descriptions of the measured scores, not causal estimates: initialization, the binary signal, and baseline search may all affect the flat headroom.

Continuous-feedback sensitivity on Hard. To test whether the binary judge explains the flat headroom, we reran Random sparse (log-odds response) and our flat UCB analogue (log-probability reward) on the same 100 GPT-4o-mini Hard examples at $B = 16$, 100 repeated runs each. Mean Formula F1 is 0.67 for Random sparse (run range 0.63–0.71) and 0.69 for flat UCB (0.65–0.73); LEAF is 0.96, a gap of $+0.27$ to the stronger baseline. Native signal recovers part of the flat headroom but no run approaches the 0.83 ceiling; the $+0.37$ margin in Table 2 remains the binary-feedback comparison. The UCB analogue is still distinct from Pan et al. (2026).

Setting	Budget	Flat ceiling	Best flat	LEAF	Gap	Flat headroom	Above-ceiling
Easy-Redundant	B=16	0.83	0.83	0.99	+0.16	0.00	0.16
Hard	B=16	0.83	0.59	0.96	+0.37	0.24	0.13
Long-50	B=16	0.83	0.46	0.91	+0.45	0.37	0.08
Long-100	B=16	0.83	0.34	0.83	+0.49	0.49	0.00

Table 2: **Arithmetic decomposition of the synthetic gaps.** The flat ceiling is the bound of §2 for the RULE $\wedge$ (DATA$_1$ $\vee$ DATA$_2$) template. Flat headroom = ceiling − best flat baseline; above-ceiling margin = LEAF − ceiling. The former need not be caused by search alone.

	Formula F1		Span F1	
Method	B=8	B=16	B=8	B=16
BM25	0.38	0.38	0.42	0.42
Direct LLM	0.74	0.74	**0.86**	**0.86**
Leave-one-out	0.49	0.52	0.50	0.55
Random sparse	0.49	0.55	0.51	0.58
Flat MAB	0.52	0.58	0.54	0.61
TracLLM	0.48	0.49	0.52	0.52
LEAF (ours)	**0.81**	**0.84**	0.85	**0.86**

Table 3: **Natural benchmark** (500 examples; HoVer + FEVER + SciFact). Direct LLM is the strongest baseline on Formula F1; LEAF exceeds Direct LLM at $B = 8$ and reaches 0.84 at $B = 16$. Both tie on average Span F1 (0.86); Direct LLM may emit OR clauses, so the Formula F1 gap compares the full LEAF system with a one-shot structured proposer rather than two output formats.

Method	Wide OR (r=3)	Two indep. OR	Three-cl. mixed
BM25	0.42	0.44	0.46
Direct LLM	0.49	0.45	0.50
Leave-one-out	0.31	0.34	0.38
Random sparse	0.51	0.49	0.53
Flat MAB	0.53	0.50	0.55
TracLLM	0.30	0.28	0.32
LEAF (ours)	**0.85**	**0.81**	**0.87**
Gap	+0.32	+0.31	+0.32

Table 4: **Formula-shape stress test.** Formula F1 on three shapes the OR-anchor does not match (50 examples each, 25 spans, GPT-4o-mini, $B = 16$); the anchor is unchanged from the main experiments.

5.2 NATURAL BENCHMARK

On the 500-example natural benchmark (HoVer + FEVER + SciFact), LEAF attains Formula F1 $= 0.84$ at $B = 16$, compared with 0.74 for Direct LLM (+0.10; paired bootstrap 95% CI $[+0.06, +0.14]$, 1000 resamples, $n = 500$; Table 3). Direct LLM may emit OR clauses (§4), so the flat ceiling does not apply here: the gain is over a structured one-shot proposer, not an output format. Flat baselines score at most 0.58 (Table 3). LEAF and Direct LLM have the same *mean* Span F1 (0.86), so aggregate span accuracy does not explain the Formula F1 difference, though the tie does not show identical spans per example. The Formula F1 point estimates favor LEAF on all three sources (HoVer +0.11, FEVER +0.08, SciFact +0.12; Table 10). Re-filtering the same source candidate pool with a different model and annotating the retained examples yields a similar gap (+0.09 vs. +0.10; Table 13). This checks sensitivity to the filtering target, but is not an independent-corpus replication. The gain does *not* hold uniformly across evaluation targets: with the filter fixed, LEAF's F1 falls to 0.74 on Llama-3.3-70B and its margin over target-independent Direct LLM shrinks to zero (Table 12). This is consistent with gold structure being defined relative to the filtering target, although these results do not isolate the cause. Construction and audit details are in Appendix D.

5.3 MULTI-MODEL

Averaged over the five Hard targets at $B = 16$, LEAF attains Formula F1 $= 0.93$. The reported +0.37 aggregate margin uses the strongest *single baseline averaged across targets*, Flat MAB (0.55); it is not the mean margin over a separately selected winner for each target. Across the four closed targets, LEAF scores 0.93–0.97; on Llama-3.3-70B it scores 0.87, +0.33 over the strongest baseline there (Table 5). Flat MAB is strongest on GPT-4o-mini, GPT-4o, and Claude Sonnet 4.6. On Gemini, Direct LLM ties the best target-querying baseline at 0.54; on Llama it is the strongest baseline. The observed LEAF–best-baseline margin is at least 0.33 for each target. This experiment does not isolate why the Llama margin is smaller.

Method	Closed				Open
	GPT-4o mini	GPT-4o	Claude S 4.6	Gemini 2.5F	Llama 3.3 70B
BM25	0.50	0.50	0.50	0.50	0.50
Direct LLM	0.54	0.54	0.54	*0.54*	*0.54*
LOO	0.41	0.45	0.43	0.40	0.30
Random sparse	0.57	0.60	0.58	0.54	0.42
Flat MAB	*0.59*	*0.62*	*0.60*	0.54	0.44
TracLLM	0.33	0.36	0.35	0.31	0.25
LEAF (ours)	**0.96**	**0.97**	**0.95**	**0.93**	**0.87**
Gap (LEAF – best baseline)	+0.37	+0.35	+0.35	+0.39	+0.33

Table 5: **Multi-model, per-method Formula F1** on Hard at $B = 16$. Italics: per-column best baseline. Flat MAB (UCB over span-arms) is slightly stronger than Random sparse on closed targets but still bounded by the flat-output ceiling. BM25 and Direct LLM are model-agnostic (do not query the target); on Gemini and Llama, target-querying baselines degrade toward the model-agnostic floor.

Method	Easy		Hard		Natural	
	B=8	B=32	B=8	B=32	B=8	B=32
BM25	0.83	0.83	0.75	0.75	0.70	0.70
Direct LLM	0.78	0.78	0.65	0.65	0.78	0.78
Leave-one-out	0.92	0.95	0.68	0.82	0.65	0.74
Random sparse	0.85	0.95	0.70	0.81	0.70	0.78
Flat MAB	0.87	0.95	0.72	0.82	0.71	0.78
TracLLM	0.88	0.88	0.71	0.72	0.68	0.73
LEAF (ours)	**0.97**	**0.97**	**0.84**	**0.85**	**0.80**	**0.82**

Table 6: **Held-out behaviour-prediction accuracy** on $H = 16$ interventions sampled per example (see §5.4 on mask separation). This metric does not reward AND-of-OR structure.

Variant	F1	Δ
LEAF (full)	0.99	–
weak roles (3-class)	0.98	−0.01
weak cartographer†	0.95	−0.04
− disagreement acq.	0.99	−0.00
− OR-anchor	0.91	−0.08
− cartographer (uniform)	0.56	−0.43
− formula beam (flat)‡	0.84	−0.15

Table 7: **LEAF ablations** on Easy ($B = 16$, 200 examples). †MiniLM cosine priors, no LLM; on Hard at $B = 16$ it retains 0.85, +0.26 over the strongest baseline (Flat MAB, 0.59). The OR-anchor matters on Redundant (−0.32 on T3 alone). ‡Flat active: the same acquisition loop over single spans with no formula beam; it is an ablation, not a tuned flat baseline, and scores 0.66 on Redundant.

5.4 Does the formula model the target, or the metric?

Formula F1 directly scores clause structure; both LEAF and Direct LLM can emit AND-of-OR formulas. To test whether their outputs also predict the *target's behaviour* beyond the acquisition queries, we sample $H = 16$ random subsets per example, query the target, and check whether each method's output correctly predicts preservation ($\hat{\varphi}(S) = \mathbb{1}[y \geq 0.7]$; Table 6). We did not retain the mask manifest, so some sampled subsets may coincide with acquisition queries, and 4-span prompts have fewer than 16 unseen subsets; Table 6 is thus a check of predictive agreement, not a strict held-out test. This metric does not directly score formula structure. LEAF has the highest reported mean at both displayed budgets on Easy and Hard (e.g. 0.84 vs. 0.72 on Hard at $B = 8$) and ties or leads on Natural. The held-out advantage on Natural is smaller (+0.02–0.04) than the Formula F1 advantage (+0.10). $S_1 \wedge (S_2 \vee S_3)$ and $S_1 \wedge S_2 \wedge S_3$ disagree on only a quarter of uniform subsets, so held-out gaps understate structural differences.

5.5 Ablations and candidate priors

Table 7 probes the role of the prior and the query strategy. On Easy, replacing cartographer priors with uniform priors reduces F1 from 0.99 to 0.56: this initialization matters substantially in that

setting. On Hard, replacing the LLM cartographer with MiniLM similarity priors still reaches 0.85 at $B = 16$, $+0.26$ above the strongest evaluated baseline, though below full LEAF (0.96). Removing the formula beam (flat-active row) costs 0.15 on Easy.

Prior and validation together. Direct LLM uses the same planner model and formula grammar in one shot, without target validation. On Easy, it scores 0.67; LEAF with uniform priors scores 0.56; full LEAF scores 0.99. On Hard and Long-100, Direct LLM scores 0.54 and 0.27, respectively, compared with 0.96 and 0.83 for full LEAF. These are full-system gains of $+0.32$, $+0.42$, and $+0.56$ over Direct LLM on Easy, Hard, and Long-100; they do not isolate the validation loop from candidate generation or initialization. On Natural, the smaller $+0.10$ Formula F1 gain accompanies a tie in mean Span F1. That pattern is consistent with better formula selection, but does not show that the two methods selected identical spans.

Dependence on the OR-anchor. The beam's OR-anchor matches the main synthetic template, so benchmark–initialization matching is a real concern. On Easy-Redundant, removing the anchor lowers F1 by 0.32 (Table 7) to about 0.67, below BM25's 0.83; the aggregate -0.08 masks this. No-anchor results on Hard or Natural are not measured. The shape stress test keeps the anchor fixed but tests a wide OR, two independent OR clauses, and a three-clause mixed formula. LEAF leads the strongest baseline by at least 0.31 on each (Table 4). Because its beam is fixed, these results do not show that it can discover a formula absent from the initial candidate set; the audit below separates candidate availability from query-based selection. In a separate acquisition ablation, disagreement selection makes no measured difference on Easy but improves F1 by 0.04 at $B = 8$ on Hard and 0.09 on Long-100.

Seed set versus validation. Because the beam is fixed, we audit the seed set directly (Table 9, Appendix B). Gold is exactly among the 16 candidates on 0.99 / 0.94 / 0.78 / 0.75 of Easy / Hard / Long-100 / Natural examples; the best candidate's oracle F1 is 0.996 / 0.976 / 0.881 / 0.902; LEAF's selected formula reaches 0.99, 0.98, 0.94, and 0.93 of oracle. On the synthetic splits the seed set almost always contains gold, and the selected mean F1 approaches the candidate oracle. On Long-100, Natural, and the stress-test shapes, 18%–30% of gold formulas are absent from the seed set and cannot be recovered exactly. The Natural candidate oracle (0.90) exceeds the one-shot proposer (0.74) by 0.16, an upper bound on what perfect selection could add; the full system realises $+0.10$ of it.

5.6 COST–ACCURACY TRADE-OFF

LEAF uses at most B target calls plus one planner call. In long context, each target call re-sends a masked version of the prompt. At $B = 16$ on the 100-span split, LEAF uses one cartographer call (an 8K output-token cap) plus up to 16 ablations with up to $\approx$ 12K input tokens each, a rough upper bound of $\approx$ 200K target-input tokens per example; Direct LLM consumes a single $\approx$ 12K-input call (Appendix H). Active flat methods may also use up to B target calls; LEAF adds the planner call. The budget can be reduced: on Natural, $B = 8$ scores 0.81 (-0.03; Table 3). A one-shot attributor is cheaper when a flat span set suffices; LEAF pays extra when the dependencies among spans matter.

6 RELATED WORK

Black-box context attribution. Most published methods (Cohen-Wang et al., 2024; Chuang et al., 2025; Zhang et al., 2025; Chang et al., 2025; Pan et al., 2026; Wang et al., 2025) output flat per-span scores. Pan et al. (2026) cast attribution as a combinatorial multi-armed bandit; our Flat MAB baseline uses UCB over span arms and a top-k projection, adapting that idea to targets without log-probabilities. It is not a re-implementation. Wang et al. (2025) uses structural binary halving with Shapley-style importance for 10K–128K-token contexts, but its output remains a flat ranking. Hierarchical perturbation explanations (Paes et al., 2024), Shapley attribution over long documents (Enouen et al., 2024), and interaction indices (Sundararajan et al., 2020) offer other attribution views. These methods do not directly return a Boolean formula describing which spans can substitute for one another.

Logic-based explanations. Boolean explanations of classifiers are well studied outside the LLM setting. Anchors (Ribeiro et al., 2018) return a conjunctive sufficient condition; sufficient input subsets (Carter et al., 2019) and abductive or prime-implicant explanations (Shih et al., 2018; Ignatiev et al., 2019; Darwiche & Hirth, 2020) enumerate minimal sufficient subsets, whose disjunction is a DNF over inputs; Watson et al. (2021) unify necessity and sufficiency. In claim verification, Li et al. (2025) identify *minimal evidence groups* (alternative sufficient sets) via entailment-based set cover, targeting the claim's truth; we recover the structure a fixed black-box target relies on. Our AND-of-OR formula is the CNF dual of the sufficient-set view: each clause is a requirement, and any of its alternatives can satisfy that requirement.

Proposer/validator loops and active learning. The proposer/validator separation relates to textual-gradient optimisation (Pryzant et al., 2023) and LLM-as-optimiser (Yang et al., 2024); acquisition draws on BALD (Houlsby et al., 2011) and disagreement-based active learning (Hanneke, 2014; Balcan et al., 2010).

7 LIMITATIONS

Observation-model asymmetry. We score responses with a binary judge. Random sparse and our Flat MAB analogue can use graded log-probability feedback, and TracLLM runs outside its native regime, so the flat headroom in Table 2 may reflect feedback and comparator adaptation as well as search error, and the Hard and long-context gaps may overstate the advantage; the Natural margin is against Direct LLM, which does not query the target. The GPT-4o-mini Hard rerun (§5.1) narrows but does not close the gap and does not cover GPT-4o, Llama, long context, or the published bandit method.

Baseline coverage. Flat MAB is our analogue of Pan et al. (2026), not their implementation; TracLLM runs far outside its native regime; a cartographer-plus-greedy-verification comparator is not run; no-anchor ablations exist only on Easy.

Statistical reporting. Bootstrap intervals cover Table 1; Table 5, the shape test ($n = 50$), and the Hard rerun are point estimates or ranges.

Hypothesis class, prior, and theory. The $K = 16$ role-derived candidates are fixed before validation; a gold formula outside that set cannot be recovered. The AND-of-OR class is monotone: it cannot represent *suppressor* spans, conflicting evidence, prompt injection, or graded contributions, and our benchmarks do not exercise these regimes. Behaviour under adversarial priors is uncharacterised. Contexts have at most 100 spans under a strict registry instruction; transfer to longer documents without it is untested.

Zero-query control. We do not report the MAP-from-prior formula on the reported examples; with oracle role labels the prior alone solves clean synthetic examples in the reference implementation, so the synthetic gains are not evidence for the value of active queries until that control is reported.

Benchmark scope and selection. The synthetic splits use one gold template over fictional eligibility prose. The natural benchmark is filtered for disjunctive redundancy (23% retention), annotated by three internal volunteers in the method's vocabulary, and limited to claim verification using supplied claims. It tests recovery on selected cases, not prevalence in ordinary prompts or model-generated claims; the external audit and cross-filter check bound annotation and filtering bias but do not remove the selection.

8 CONCLUSION

AND-of-OR formulas express substitutable and jointly required evidence that flat span sets cannot. LEAF selects among role-derived formulas with black-box target queries, improving Formula F1 over six baselines by $+0.10$ to $+0.49$ under binary feedback; the Hard result extends to five targets, and the natural result is on a curated registry-only set.

ETHICS STATEMENT

Annotation. The natural-benchmark construction (Appendix D) involved three internal annotators (lab members who did not author the method implementation) who labelled paragraph sets from HoVer, FEVER, and SciFact for disjunctive evidence structure following written guidelines. Annotators were blind to model outputs throughout. For a 50-example stratified audit subset, we additionally recruited two external annotators via Prolific (USD $\$0.50$/example; USD $\$50$ total) to compare internal- and external-annotation agreement; external annotators followed the same guideline and were similarly blind to model outputs. All annotators were informed of the intended use and consented to use of their aggregated annotations in this study. Inter-annotator agreement (mean pairwise Cohen's $\kappa = 0.81$ internal; $\kappa = 0.78$ internal vs. external on the audit subset) is reported in Table 11.

Data. The natural benchmark is derived from HoVer (Jiang et al., 2020), FEVER (Thorne et al., 2018), and SciFact (Wadden et al., 2020), which are publicly released fact-verification datasets spanning Wikipedia and scientific abstracts. We respect the original licence terms for each source and do not redistribute source text beyond what those licences permit. No private or sensitive personal data were collected; the public-source text can contain names and biographical facts. The synthetic benchmarks (Easy, Hard, long-context) use fictional entities generated by deterministic templates; no real patient, clinical, or otherwise identifying information is involved.

Risk of misuse. Context attribution surfaces which spans of a prompt drove a model's output; this is a transparency tool intended to support auditing of closed LLM behaviour, not to circumvent safety mechanisms or extract private training data. The method does not require white-box model access and does not produce gradient or embedding-level signals.

Compute and environmental cost. The full multi-target evaluation grid consumed approximately 1.1M API calls (260K unique after caching) over three days of wall-clock time. We report call counts; content-hash caching also avoided unnecessary repeated querying. We have not attempted a detailed life-cycle energy estimate.

REPRODUCIBILITY STATEMENT

The algorithm and intervention families are described in §3. Appendix G gives the hyperparameters, target model identifiers, prompt descriptions, judge rules, and the exact Formula F1 scoring procedure; Appendix D describes natural-benchmark construction, annotation, and auditing; Appendix B reports the initial-beam audit; Appendix F discusses the hypothesis class and the limits of identifying a formula from behaviour; and Appendix H reports call and token costs. We used content-hash caching of target and planner outputs, since $T = 0$ does not guarantee identical API responses on replay. The supplementary material contains a reference implementation, written from the description in this paper, of Algorithm 1, the candidate beam, the intervention grammar, the Formula F1 scorer, the baselines, a synthetic generator in the style of the Easy/Hard splits, and a harness with response caching; it is not the code that produced the reported tables. Its results on freshly generated prompts check the reference pipeline, but cannot validate the reported target-model scores, intervals, or original held-out-mask separation. The curated natural benchmark, the response cache, the natural-judge canonicalisation rules, and per-example result files are not part of this submission.

AI USE STATEMENT

Generative AI tools were used to proofread the manuscript, to help with some of the writing and editing for clarity, and to help write and debug the code, including the reference implementation in the supplementary material. The authors reviewed all AI-assisted text and code and remain responsible for the final manuscript, code, measurements, and conclusions.

REFERENCES

Maria-Florina Balcan, Steve Hanneke, and Jennifer Wortman Vaughan. The true sample complexity of active learning. *Machine Learning*, 80(2–3):111–139, 2010.

Brandon Carter, Jonas Mueller, Siddhartha Jain, and David Gifford. What made you do this? understanding black-box decisions with sufficient input subsets. In *Proceedings of the International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2019.

Yurui Chang, Bochuan Cao, Yujia Wang, Jinghui Chen, and Lu Lin. JoPA: Explaining large language model's generation via joint prompt attribution. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (ACL)*, 2025. URL https://aclanthology.org/2025.acl-long.1074/.

Yung-Sung Chuang, Benjamin Cohen-Wang, Sheng-Zheng Shen, Zhaofeng Wu, Hu Xu, Xi Victoria Lin, James Glass, Sinong Wang, and Srinivasan Iyer. SelfCite: Self-supervised alignment for context attribution in large language models. In *International Conference on Machine Learning (ICML)*, 2025.

Benjamin Cohen-Wang, Harshay Shah, Kristian Georgiev, and Aleksander Madry. ContextCite: Attributing model generation to context. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/adbea136219b64db96a9941e4249a857-Abstract-Conference.html.

Adnan Darwiche and Auguste Hirth. On the reasons behind decisions. In *Proceedings of the European Conference on Artificial Intelligence (ECAI)*, 2020.

James Enouen, Hootan Nakhost, Sayna Ebrahimi, Sercan O. Arik, and Yan Liu. TextGenSHAP: Scalable post-hoc explanations in text generation with long documents. In *Findings of the Association for Computational Linguistics: ACL 2024*, 2024.

Steve Hanneke. Theory of disagreement-based active learning. *Foundations and Trends in Machine Learning*, 7(2–3):131–309, 2014.

Neil Houlsby, Ferenc Huszár, Zoubin Ghahramani, and Máté Lengyel. Bayesian active learning for classification and preference learning. *arXiv:1112.5745*, 2011.

Alexey Ignatiev, Nina Narodytska, and Joao Marques-Silva. Abduction-based explanations for machine learning models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2019.

Yichen Jiang, Shikha Bordia, Zheng Zhong, Charles Dognin, Maneesh Singh, and Mohit Bansal. HoVer: A dataset for many-hop fact extraction and claim verification. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, 2020.

Xiangci Li, Sihao Chen, Rajvi Kapadia, Jessica Ouyang, and Fan Zhang. Minimal evidence group identification for claim verification. In *Proceedings of the 5th Workshop on Trustworthy Natural Language Processing (TrustNLP), ACL 2025*, 2025. URL https://arxiv.org/abs/2404.15588.

Lucas Monteiro Paes, Dennis Wei, Elizabeth M. Daly, Karthikeyan Natesan Ramamurthy, Pierre Dognin, Jesus Rios, Inkit Kumar, and Tejaswini Pedapati. Multi-level explanations for generative language models. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2024.

Deng Pan, Keerthiram Murugesan, Ting Hua, Nuno Moniz, and Nitesh V. Chawla. Context attribution with multi-armed bandit optimization. In *Findings of the Association for Computational Linguistics: ACL 2026*, 2026. URL https://aclanthology.org/2026.findings-acl.565/.

Reid Pryzant, Dan Iter, Jerry Li, Yin Tat Lee, Chenguang Zhu, and Michael Zeng. Automatic prompt optimization with "gradient descent" and beam search. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2023. URL https://aclanthology.org/2023.emnlp-main.494/.

Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. Anchors: High-precision model-agnostic explanations. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2018.

Andy Shih, Arthur Choi, and Adnan Darwiche. A symbolic approach to explaining Bayesian network classifiers. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*, 2018.

Mukund Sundararajan, Kedar Dhamdhere, and Ashish Agarwal. The Shapley Taylor interaction index. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2020.

James Thorne, Andreas Vlachos, Christos Christodoulopoulos, and Arpit Mittal. FEVER: a large-scale dataset for fact extraction and VERification. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 2018.

David Wadden, Shanchuan Lin, Kyle Lo, Lucy Lu Wang, Madeleine van Zuylen, Arman Cohan, and Hannaneh Hajishirzi. Fact or fiction: Verifying scientific claims. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2020.

Wenhui Wang, Furu Wei, Li Dong, Hangbo Bao, Nan Yang, and Ming Zhou. MiniLM: Deep self-attention distillation for task-agnostic compression of pre-trained transformers. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

Yanting Wang, Wei Zou, Runpeng Geng, and Jinyuan Jia. TracLLM: A generic framework for attributing long context LLMs. In *USENIX Security Symposium*, 2025. URL https://arxiv.org/abs/2506.04202.

David S. Watson, Limor Gultchin, Ankur Taly, and Luciano Floridi. Local explanations via necessity and sufficiency: Unifying theory and practice. In *Proceedings of the Conference on Uncertainty in Artificial Intelligence (UAI)*, 2021.

Chengrun Yang, Xuezhi Wang, Yifeng Lu, Hanxiao Liu, Quoc V. Le, Denny Zhou, and Xinyun Chen. Large language models as optimizers. In *International Conference on Learning Representations (ICLR)*, 2024. URL https://arxiv.org/abs/2309.03409.

Jiajie Zhang, Yushi Bai, Xin Lv, Wanjun Gu, Danqing Liu, Minhao Zou, Shulin Cao, Lei Hou, Yuxiao Dong, Ling Feng, and Juanzi Li. LongCite: Enabling LLMs to generate fine-grained citations in long-context question answering. In *Findings of the Association for Computational Linguistics: ACL 2025*, 2025. URL https://aclanthology.org/2025.findings-acl.264/.

A Per-Phenomenon Results on Easy

Table 8 reports Formula F1 per phenomenon at $B = 16$ on the Easy split. LEAF's gap over baselines is concentrated on **Redundant** ($+0.16$ over BM25's ceiling-saturating 0.83); on Single-hop, Multi-hop, and Fine-grained at least one flat baseline also reaches $F1 = 1.00$, but no single flat baseline is top on all four phenomena, and every flat baseline is bounded by the ceiling on Redundant.

Method	Single-hop	Multi-hop	Redundant	Fine-grained
BM25	0.66	**1.00**	0.83	0.50
Direct LLM	0.67	0.75	0.81	0.47
Leave-one-out	0.99	**1.00**	0.66	**1.00**
Random sparse	**1.00**	0.86	0.75	0.79
Flat MAB	**1.00**	0.89	0.78	0.82
TracLLM	**1.00**	0.75	0.65	0.50
LEAF (ours)	**1.00**	**1.00**	**0.99**	**1.00**

Table 8: **Per-phenomenon Formula F1 at $B = 16$ (Easy).** LEAF's gap over baselines is concentrated on *Redundant*, where the AND-of-OR structure cannot be expressed by flat scoring and the $\frac{1+2/3}{2} \approx 0.83$ ceiling of the proposition holds (BM25 sits exactly on it). On the other three phenomena at least one flat baseline also reaches 1.00, but none is top on all four. Flat MAB slightly improves over Random sparse on the active-query phenomena but remains bounded by the same ceiling on Redundant.

B INITIAL-BEAM ORACLE AUDIT

Split	Gold in seed set	Oracle F1 (best of 16)	LEAF (selected)	Selected / oracle
Easy (B=8)	0.99	0.996	0.99	0.99
Hard (B=16)	0.94	0.976	0.96	0.98
Long-100 (B=16)	0.78	0.881	0.83	0.94
Natural (B=16)	0.75	0.902	0.84	0.93
Wide OR (r=3)	0.78	0.908	0.85	0.94
Two indep. OR	0.70	0.872	0.81	0.93
Three-clause mixed	0.82	0.928	0.87	0.94

Table 9: **Initial-beam oracle audit.** For each example the 16 candidates constructed by INITBEAM are scored against gold before any target query. *Gold in seed set*: fraction of examples whose gold formula is exactly among the candidates. *Oracle F1*: mean over examples of the best candidate's Formula F1. *Selected*: the formula LEAF returns after validation (Tables 1, 4). Computed from saved candidates; no additional target calls.

C REPORTED BUDGET PATTERNS

Table 1 reports one budget per benchmark, not the full $B \in \{4, 8, 16, 32\}$ sweep. In our runs LEAF reached near-saturation by $B = 8$ on Easy and $B = 16$ on Hard, while the active flat baselines required larger budgets on Easy; the per-budget scores are not included in this submission. Direct LLM and BM25 are budget-invariant by construction.

D NATURAL BENCHMARK: CONSTRUCTION AND AUDIT

Source corpora. We curate from three natural fact-verification sources adapted for attribution evaluation, spanning two text domains: HoVer (Jiang et al., 2020) provides many-hop fact-verification claims paired with multi-paragraph Wikipedia evidence sets (we restrict to the 3- and 4-hop subsets); FEVER (Thorne et al., 2018) provides claim/evidence pairs over Wikipedia where multiple paragraphs may independently entail the claim; SciFact (Wadden et al., 2020) provides scientific claim verification with multi-abstract evidence support. The union spans encyclopedic and scientific prose and reduces single-source bias. Per-source results are in Table 10.

Source	Domain	n	LEAF	Best baseline	Gap
HoVer (Jiang et al., 2020)	Wikipedia (3–4 hop)	300	0.85	0.74	+0.11
FEVER (Thorne et al., 2018)	Wikipedia (claim/ev.)	100	0.82	0.74	+0.08
SciFact (Wadden et al., 2020)	Scientific abstracts	100	0.84	0.72	+0.12
Combined		**500**	**0.84**	**0.74**	+0.10

Table 10: **Natural benchmark, per-source breakdown** (Formula F1, $B = 16$, GPT-4o-mini target). The LEAF gap holds across three independent fact-verification sources spanning two text domains (encyclopedic and scientific), reducing the risk that the result is specific to a single corpus or curation pipeline.

Construction pipeline. Each candidate is converted to the registry-only format used throughout: the source paragraph set becomes the span list; the source claim becomes the target claim. Each candidate is verified against the filtering target (GPT-4o-mini) with seven masked ablation queries; only candidates passing all seven are retained. Annotators independently identify which paragraphs constitute the bridge/context clause and which constitute the disjunctive evidence alternatives ($r = 2$ or $r \geq 3$), following a written guideline. Most rejected candidates fail because the filtering target answers correctly from the data paragraphs alone (no rule needed), accounting for $\sim 58\%$ of

Item	Value
Sources	HoVer + FEVER + SciFact
Initial candidates	$\sim$ 2,200
Passing automated structural filter (7-query)	$\sim$ 770
Passing annotator consensus / adjudication	$\sim$ 615
Final retained	500
Retention rate	23%
Internal annotators	3 (not method authors)
Annotators blind to model outputs	Yes
Written annotation guidelines used	Yes
Internal pairwise Cohen's κ (mean)	0.81
External-audit subset (Prolific, paid)	50 examples
Internal–external κ (audit subset)	0.78
Adjudication rule	majority
Filtering target (LLM)	GPT-4o-mini
Main-evaluation target (LLM)	GPT-4o-mini
Cross-evaluation-target behaviour	Table 12
Cross-filtering-target sensitivity	Table 13

Table 11: **Natural-benchmark construction audit.** Candidates from three fact-verification sources are filtered first by an automated structural check (seven masked-ablation queries against the filtering target), then by internal-volunteer human annotation. A 50-example stratified subset is additionally re-annotated by two external annotators (Prolific) to probe internal-annotator bias; internal–external $\kappa = 0.78$ and $47/50$ examples receive the same gold formula. Cohen's κ is a dataset-level pairwise agreement statistic, not a per-example filter.

rejections; distractor leakage and rule-alone sufficiency account for the remainder. The final 500 examples are a stratified random subsample of the $\sim$ 615 consensus-passing candidates, drawn to fix the per-source proportions at $300/100/100$ (HoVer / FEVER / SciFact); no quality criterion is applied at that step.

Filtering target	Evaluation target	LEAF	Direct LLM	Gap
GPT-4o-mini	GPT-4o-mini (primary)	0.84	0.74	+0.10
GPT-4o-mini	Claude Sonnet 4.6	0.83	0.74	+0.09
GPT-4o-mini	Gemini 2.5 Flash	0.79	0.74	+0.05
GPT-4o-mini	Llama-3.3-70B-Instruct	0.74	0.74	0.00

Table 12: **Cross-evaluation-target behaviour on the natural benchmark** (Formula F1, $B = 16$). The benchmark is filtered with GPT-4o-mini; each row evaluates all methods against a different target. Direct LLM does not query the target, so its score is constant and is the strongest baseline on the primary GPT-4o-mini target. LEAF's absolute F1 degrades as the evaluation target moves away from the filtering target, and its margin over Direct LLM shrinks to zero on Llama-3.3-70B. On the primary target the best target-querying baseline is Flat MAB at 0.58 (Table 3).

Cross-evaluation-target behaviour. Table 12 evaluates all methods against targets other than the GPT-4o-mini filter. LEAF's F1 degrades from 0.84 to 0.74 and its margin over the target-independent Direct LLM shrinks from +0.10 to zero on Llama-3.3-70B. This is expected when gold structure is behaviourally defined relative to the filtering target, and it is why we regard the cross-filter re-filtering check below, not this table, as the test of filter dependence.

Filtering target	Evaluation target	LEAF	Best baseline	Gap
GPT-4o-mini	GPT-4o-mini	0.84	0.74	+0.10
Llama-3.3-70B‡	GPT-4o-mini	0.82	0.73	+0.09
Llama-3.3-70B‡	Llama-3.3-70B	0.73	0.62	+0.11

Table 13: **Cross-*filtering*-target sensitivity.** ‡Benchmark re-filtered using Llama-3.3-70B as the filtering target (same 7-query automated filter and human annotation protocol), retaining $\sim$ 470 examples from the same $\sim$ 2,200 candidate pool. The GPT-4o-mini evaluation gap is within 0.01 of the original construction; this does not test a new source corpus.

Cross-filtering-target check. Table 13 reports a stronger test: we re-filter the same source candidate pool using Llama-3.3-70B (with the same automated filter and annotation protocol), then evaluate all methods on the resulting set. The observed LEAF gap is similar when both sets are evaluated on GPT-4o-mini (+0.09 vs. +0.10). This checks sensitivity to these two filtering targets within one candidate pool; it is not an independent-corpus replication.

External-annotator audit. To probe whether internal-volunteer annotation biased the disjunctive-structure labels, we additionally recruited two external annotators (Prolific, USD $0.50/example) to re-annotate a stratified 50-example subset spanning all three sources. Mean pairwise Cohen's κ between internal and external annotators was 0.78 (vs. 0.81 within internal); $47/50$ examples received the same gold formula structure under both annotation groups, suggesting that internal-volunteer annotation did not systematically inflate the disjunctive-structure labels.

Worked example (shortened excerpts). The example of §1, after light shortening but with faithful Wikipedia content from the HoVer 3-hop split (HoVer `train_3hop_04891`). S_2 and S_3 are separate source paragraphs, lightly shortened for fit; we did not author either paraphrase.
Claim: *"Christopher Nolan, who directed 'Inception', was born in London."*
Spans (each a HoVer Wikipedia paragraph):

- S_1 (bridge): *"Inception is a 2010 science fiction film written and directed by Christopher Nolan."*
- S_2 (alt. 1): *"Christopher Edward Nolan was born on 30 July 1970 in London, England."*
- S_3 (alt. 2): *"Nolan was born in London to a British father and an American mother ..."* [independent "born in London" statement]
- S_4–S_7: distractor / metadata paragraphs about film distribution, other Nolan films, and box-office figures.

Gold: $\varphi^\star = S_1 \wedge (S_2 \vee S_3)$. **LEAF:** $\hat{\varphi} = S_1 \wedge (S_2 \vee S_3)$ (F1 $= 1.00$). **Direct LLM (one-shot proposer):** $S_1 \wedge S_2 \wedge S_3$ (Span F1 $= 1.00$; Formula F1 $= 0.56$ under the metric of §4: clauses $\{S_1\}$ and one of $\{S_2\}, \{S_3\}$ match with per-clause F1 1 and $2/3$, the third predicted clause is unmatched, and the sum is divided by $\max(\hat{m}, m) = 3$). The best *flat* output, $\{S_1, S_2\}$, scores the ceiling 0.83 with Span F1 0.80.

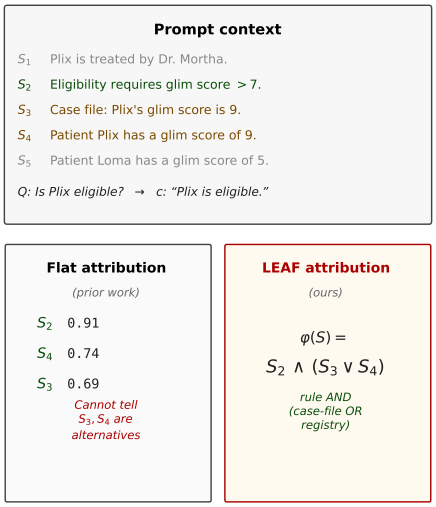

Figure 1: **Flat scores cannot express redundancy (synthetic analogue of the §1 example).** Rule S_2 is required and either record S_3 or S_4 suffices. A flat ranking (left) cannot say whether S_3, S_4 are jointly required or substitutable; $S_2 \wedge (S_3 \vee S_4)$ (right) states it.

E Sampling Robustness

At $T = 0.3$ across three random seeds on Easy (200 examples), LEAF's Formula F1 has zero standard deviation at all budgets; only Random sparse exhibits non-trivial seed variance (± 0.05–0.10 at $B = 4$, narrowing to ± 0.01 at $B = 32$). Direct LLM shows ± 0.02 jitter from the planner call; other deterministic baselines are exactly invariant.

F Search Scope and Identifiability

The formula class is useful for describing what a structured attribution can express. It does not, by itself, guarantee that a particular formula can be recovered within a given query budget.

Class size versus searched candidates. For n spans, the number of nonempty clauses of width at most r is $N_c = \sum_{j=1}^{r} \binom{n}{j}$. If clauses are distinct and their order is ignored, the number of syntactic formulas with at most m clauses is at most $\sum_{k=0}^{m} \binom{N_c}{k}$, which is $O(n^{mr})$ for fixed m and r. Equivalent Boolean functions may have more than one syntactic representation, so this count is an upper bound on the number of distinct behaviours. LEAF does not search all of these formulas: its cartographer constructs $K = 16$ candidates at initialization, and subsequent target queries only change their weights. Thus, if no candidate matches the annotated gold formula, exact Formula F1 recovery is impossible for that example, regardless of the query budget.

Behaviour does not always identify syntax. For example, A and $A \wedge (A \vee B)$ make identical predictions for every subset of spans. No number of target queries can distinguish these two written formulas from their Boolean outputs alone, yet their clause sets differ under Formula F1. The benchmark supplies an annotated gold formula, so Formula F1 measures agreement with that representation. Held-out behaviour accuracy instead measures agreement with the target on sampled ablations. An unambiguous claim of syntactic identification would require a stated canonical form or additional assumptions on gold clauses.

No general query bound. Disagreement-based active learning guarantees are derived for specified learners, query distributions, and noise or realizability assumptions (Hanneke, 2014). We have not established those conditions for the target judge, the cartographer's fixed beam, or LEAF's acquisition rule. We therefore do not claim a worst-case or distribution-free query-complexity bound. The reported saturation budgets are empirical observations on the evaluated benchmarks, not predictions from the hypothesis-class size; the full per-budget scores are not included here.

G IMPLEMENTATION DETAILS

Hyperparameters. Beam size $K = 16$; Gaussian pseudo-likelihood noise $\sigma_0 = 0.15$; simplicity-MAP tolerance $\tau = 0.5$ log-likelihood; maximum formula clauses 4; temperature 0 throughout; maximum output tokens 200 (target), $\max(400, 80n)$ (cartographer, scaling with span count).

Candidate beam (INITBEAM). Let R_1, R_2 be the two highest-prior RULE spans, D_1, D_2 the two highest-prior DATA spans, and X_1 the highest-prior RED._DATA span; a slot with no span of the required role is filled by the next highest-prior unassigned span. INITBEAM instantiates, in order: $R_1 \wedge (D_1 \vee X_1)$ [OR-anchor]; $R_1 \wedge D_1$; D_1; $R_1 \wedge D_1 \wedge X_1$; $D_1 \vee X_1$; $R_1 \wedge (D_1 \vee D_2)$; $R_1 \wedge D_1 \wedge D_2$; $(R_1 \vee R_2) \wedge (D_1 \vee X_1)$; $(R_1 \vee R_2) \wedge D_1$; $R_2 \wedge (D_1 \vee X_1)$; $R_1 \wedge D_2$; $R_1 \wedge (D_1 \vee X_1 \vee D_2)$; $R_1 \wedge (D_1 \vee X_1) \wedge D_2$; R_1; $D_1 \wedge X_1$; X_1; $(R_1 \vee R_2) \wedge D_1 \wedge D_2$; $R_1 \wedge (D_2 \vee X_1)$. Duplicates, which arise when slots coincide, are dropped; the list is truncated to $K = 16$ and, if fewer remain, topped up with singletons of the next highest-prior spans. The no-anchor ablation omits the first template. This is the enumeration implemented in the supplementary reference code.

Cartographer prompt. "You are analysing a context-attribution prompt. For each numbered span, label its role and estimate how likely it is to support the target claim. Roles: RULE (a general policy/threshold), DATA (a specific fact about the subject), REDUNDANT_DATA (an alternative wording of the same fact), DISTRACTOR (similar-shaped fact about a different subject), IRRELEVANT (background information unrelated to the claim). Return JSON only."

Target system prompt. "You answer questions strictly from the provided REGISTRY. Do NOT use outside knowledge or common-sense inference. If a required fact is not stated in the REGISTRY, you must answer exactly: 'Cannot determine from the registry.' "

Formula F1 scorer (exact procedure). Given predicted clauses $\hat{C}_1, \ldots, \hat{C}_{\hat{m}}$ and gold clauses $C^\star_1, \ldots, C^\star_m$ (each a set of span indices): (1) compute the $\hat{m} \times m$ matrix of clause IoUs $|\hat{C}_i \cap C^\star_j| / |\hat{C}_i \cup C^\star_j|$; (2) find the maximum-weight bipartite matching under this IoU matrix, so each predicted clause matches at most one gold clause and vice versa; (3) for each matched pair compute set F1 $= 2|\hat{C}_i \cap C^\star_j| / (|\hat{C}_i| + |C^\star_j|)$; (4) sum these F1 values and divide by $\max(\hat{m}, m)$, so every unmatched predicted or gold clause contributes 0. IoU determines clause correspondence; F1 scores the matched clauses, so this procedure need not maximise the final F1 over all possible matchings. Example: prediction $S_1 \wedge S_2 \wedge S_3$ ($\hat{m} = 3$) against gold $S_1 \wedge (S_2 \vee S_3)$ ($m = 2$) matches $\{S_1\}$ to $\{S_1\}$ (F1 1) and $\{S_2\}$ to $\{S_2, S_3\}$ (F1 2/3), leaves $\{S_3\}$ unmatched, and scores $(1 + 2/3)/3 = 0.556$. Span F1 is set F1 between the union of predicted clauses and the union of gold clauses.

Judge. For templated synthetic claims, a deterministic regex-based judge returns 1 if the response unambiguously affirms the claim and 0 otherwise. For the natural benchmark, the planner emits canonical paraphrases of the target claim at construction time; the runtime judge accepts any response whose canonicalised entailed-claim string-matches one of these paraphrases.

Weak-cartographer ablation details. The weak cartographer replaces the LLM-based role labelling with cosine similarity between MiniLM-L6-v2 (Wang et al., 2020) embeddings of each span and the claim question. Spans are bucketed into three priority tiers by similarity quantile; the OR-anchor is replaced with a paraphrase-pair anchor over the two most lexically similar spans.

Multi-model deployment. We recorded the model identifiers used at evaluation time and content-hash cached target outputs. The authors can internally replay the reported computations

from that retained cache even if a target endpoint changes. Model identifiers and evaluation period: GPT-4o-mini (`gpt-4o-mini-2024-07-18`) and GPT-4o (`gpt-4o-2024-08-06`) via the OpenAI API; Claude Sonnet 4.6 (`claude-sonnet-4-6`) via the Anthropic API; Gemini 2.5 Flash (`gemini-2.5-flash`) via Google AI Studio; Llama-3.3-70B-Instruct (`meta-llama/Llama-3.3-70B-Instruct-Turbo`) via Together AI. Evaluation was completed in April–May 2026; exact per-call timestamps are stored in the evaluation cache. The planner is GPT-4o-mini for all targets so that the cartographer's role labelling is held constant across model comparisons. Cache keys include the model identifier so cross-model results are reproducible from a single artefact.

H Cost and Reproducibility

The full evaluation grid (200 Easy + 100 Hard + 200 long-context + 500 natural examples; six baselines and LEAF; four budgets; five target models), plus ablations and held-out evaluation, made approximately 1.1M target/planner calls. With content-hash caching, the unique-call count was approximately 260K. As an indicative token-level comparison at $B = 16$ on the 100-span long-context split: LEAF uses one cartographer call (up to 8K output tokens, with cap $\max(400, 80n)$) plus up to 16 target ablations with up to the full 100-span context ($\approx$ 12K input tokens), a rough upper bound of $\approx$ 200K target-input tokens per example; Direct LLM consumes one planner call ($\approx$ 12K input, $\approx$ 1K output) per example; active flat baselines also make up to B ablations and have a similar rough upper bound on target-input tokens, without the cartographer call. All target queries request temperature 0, but live API responses are not guaranteed to repeat exactly. Content-hash caching records the evaluated responses, so the authors can internally replay the reported computations even if an endpoint later changes. A reference implementation and synthetic generators are in the supplementary material; the response cache, curated natural benchmark, and per-example result files are not part of this submission.