BELONG: REPOSITORY EVIDENCE IMPROVES GO JSON CHOICE INFERENCE

Anonymous authors
Paper under double-blind review

ABSTRACT

Code can compile and pass its tests while violating the repository's established serialization convention because the missing constraint lives in nearby precedents. We introduce BELONG, a natural-label held-out benchmark for repository-conditioned Go JSON choice inference. Its source census contains 37,314 decisions from 8 public repositories. On 100 convention-ambiguous decisions with sparse local context, three language-model families predict both field name and presence policy under seven controlled evidence views. On this deliberately repository-balanced sparse-local stress sample, Original Raw raises semantic choice accuracy from 60.0% to 93.3%: +33.3 points [26.3, 40.7], with 107 rescues and 7 regressions. Exclusive Raw removes every target-file row, exact-field row, and target-identifier mention yet retains +21.7 points [15.0, 28.7]. A closest-available matched control supplies equally long and filtered evidence from other repositories; Exclusive Raw remains +19.3 points ahead [13.0, 25.7]. Original Raw therefore measures repository-assisted reconstruction; the no-copy contrast estimates the complete same-repository evidence construction rather than repository identity alone. After removing 20 targets whose non-JSON tags expose the name, Original Raw and the matched no-copy comparison retain +30.4 and +17.1 joint points, with both name and presence intervals above zero. On a separate 28-task holdout from four repositories created after the dated DeepSeek model identifier, a labeled exploratory Original Raw condition adds +16.1 points [1.8, 32.1]. The null no-copy contrast localizes this transfer to concrete precedents. Across 6,576 exact-label predictions costing $7.706, BELONG provides a reproducible, training-free diagnostic of repository evidence for its evaluated Go JSON populations.

1 INTRODUCTION

Suppose a code model operating inside a coding agent sees a Go field named `PlaceOfSupplyScheme`. Converting the identifier to `place_of_supply_scheme` is straightforward. Choosing `omitempty` requires repository usage evidence. In one real repository, the same field appears in several request types with both policies. The repository's concrete usages resolve the ambiguity left by the masked local struct and its comments. Three model families all select the wrong presence policy from local context and all recover the developer's choice after seeing retrieved precedents.

This is a common form of underspecification. Tests and compilers establish functional acceptability, yet leave naming, serialization, construction, and error-handling choices open. Coding agents evaluated only by task completion can therefore produce plausible patches that do not belong in the repository. Repository-level code models already use cross-file retrieval to improve completion and repair (Shrivastava et al., 2023; Zhang et al., 2023; Wu et al., 2024), but current benchmarks rarely isolate whether a model inferred the local choice or simply produced functionally adequate code. We isolate this capability as a controlled code-model subproblem inside repository agents: retrieval is fixed, tools are absent, and only the final repository-conditioned choice is scored.

BELONG turns local choice inference into an exact-label evaluation. We mine natural serialization sites from pinned repository snapshots, mask the complete JSON decision, and retain the developer's tag as ground truth. Each task exposes the field, type, file, and an 11-line local window. Controlled conditions then add either prescriptive choice cards, descriptive repository profiles, similarity-retrieved

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.

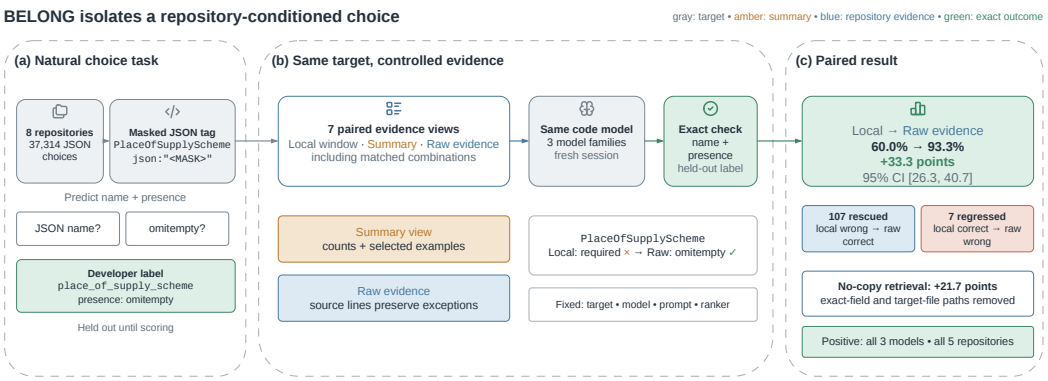

Figure 1: BELONG isolates repository-conditioned choice inference. The target, model, prompt, and ranker stay fixed; only repository evidence changes. Raw evidence adds 33.3 points, and evidence with every direct target path removed still adds 21.7.

source lines, or matched combinations. The same target is answered in a fresh session under every condition, so paired transitions directly measure what repository evidence changes.

Initial raw precedents reach 93.3% versus 60.0% Local across three model families. Equal-length foreign evidence remains near Local; fresh repeats retain 33.3–37.7-point gains; and unbatched calls retain 28.9 points. Full no-copy retrieval adds 21.7 points and beats its matched foreign control by 19.3 [13.0, 25.7]. GPT-5.6 Sol retains a 20-point gap, while family-weighted and handwritten-only analyses remain positive. A new-repository holdout then localizes transfer: concrete precedents add 16.1 points [1.8, 32.1]. Filtered convention evidence yields a null effect on these 28 tasks, defining the measured boundary: repository-assisted reconstruction transfers on this four-repository holdout, while no-copy abstraction remains population-dependent.

Our contributions are:

- **A natural serialization-choice benchmark.** BELONG contributes a 37,314-decision census across 8 repositories, a disjoint sparse-local confirmation set, exact labels, seven paired context conditions, and complete model/cost records.
- **A controlled representation study.** Over 2,100 main predictions, raw repository evidence adds 33.3 semantic accuracy points [26.3, 40.7] over local context and improves every evaluated model and repository.
- **A decisive robustness suite.** Original- and Exclusive-Raw-matched foreign controls, two full repeats, an outcome-blind unbatched replication, a frontier-model control, and a post-release repository holdout show when same-repository evidence improves naming and presence beyond token budget, run variation, batching, model tier, and direct target paths—and where it stops.

2 REPOSITORY-CONDITIONED CHOICE INFERENCE

2.1 TASK DEFINITION

A repository snapshot contains decisions $d_i = (x_i, y_i)$, where x_i is a code site and $y_i = (n_i, o_i)$ is a JSON field name and a Boolean presence policy. We remove the complete JSON tag from x_i and expose a local view L_i plus optional repository evidence E_i. A model predicts $\hat{y}_i = f(L_i, E_i)$. The primary metric is *semantic choice accuracy*: both the exact field name and `omitempty` decision must match. The parser accepts a redundant `,omitempty` suffix inside the predicted name when the separate Boolean agrees; strict protocol accuracy, which does not normalize that redundancy, is reported alongside the primary metric in the artifacts.

Masking removes the complete JSON tag but preserves the natural declaration, including other struct tags. We therefore audit whether any non-JSON tag on that declaration contains the exact held-out

Repository	Decisions	Multiword	Dominant multiword	Omit %
controller-runtime	42	14	camel (14)	28.6
echo	9	0	—	22.2
fiber	106	69	snake (68)	2.8
go-github	5,935	2,805	snake (2,570)	81.5
go-redis	53	24	snake (23)	34.0
grpc-gateway	2,090	1,467	camel (1,060)	89.2
kubernetes-api	3,673	1,688	camel (1,635)	71.2
stripe-go	25,406	11,695	snake (11,410)	63.7

Table 1: The BELONG census. "Multiword" includes fields for which snake and lower camel transforms differ; custom choices remain in the counts.

name and report all components again after removing such sites. This distinguishes naming inference from presence inference without changing a single model call.

The label is a held-out developer decision rather than a generated preference or an LLM judgment. This makes every prediction automatically checkable. It also captures the central property of a convention: the answer is conditioned on a particular repository, not on a universal style rule. The metric measures fidelity to a historical choice, not whether that choice is the uniquely best API design. Repeated generated decisions can encode a stable schema convention, but not independent developer judgments; we therefore report handwritten and generated strata separately.

2.2 NATURAL DECISION CENSUS

We pin public repository commits and parse non-test, non-vendored Go files for struct fields with JSON tags. Each record stores repository and commit, file and line, package, field and type, exact tag, presence policy, source line, and an 11-line masked window. The resulting census contains 37,314 decisions from eight repositories. Table 1 shows the variation that makes repository conditioning useful. Fiber and Stripe use snake case for nearly every multiword field; Kubernetes API uses lower camel case; gRPC-Gateway contains substantial populations of both plus custom names. Presence rates range from 2.8% to 89.2%.

3 SEVEN VIEWS OF THE SAME REPOSITORY EVIDENCE

For every target, a deterministic retriever ranks other decisions in the repository by identifier-token overlap, name similarity, type agreement, and package agreement. The target itself is always removed. For target t and candidate c, the exact score is

$$s(t,c) = 1.8J(W(f_t), W(f_c)) + \mathrm{seq}(f_t, f_c) + 0.45\mathbb{1}[b_t = b_c] + 0.25\mathbb{1}[p_t = p_c],$$

where f, b, and p are the visible Go field identifier, type bucket, and package; W tokenizes the identifier, J is Jaccard overlap, and seq is normalized sequence similarity. Neither the hidden JSON name nor the `omitempty` label enters retrieval. A released invariance test mutates both gold labels for every target and verifies identical scores and rankings. We render seven views:

1. **Local**: repository, path, field, type, and the masked 11-line window.
2. **Choice Card**: a prescriptive name and presence recommendation, support counts, three supporting examples, and the nearest exception.
3. **Choice Profile**: descriptive repository/package style counts, type-conditioned presence counts, eight comparable precedents, and up to three counter-precedents.
4. **Card + raw**: the card followed by raw retrieved source lines.
5. **Raw evidence**: source lines alone, expanded to match Card + raw.
6. **Profile + raw**: the descriptive profile followed by raw source lines.
7. **Raw evidence (matched)**: source lines alone, expanded to match Profile + raw. Mean batched inputs are 27,151 versus 27,343 characters, a 0.7% difference.

Selection stage	Remaining
Natural decisions in census	37,314
Disjoint from pilot	37,230
Ambiguous multiword/custom transform	17,678
Supported target label family	17,561
Sparse local window (zero visible JSON tags)	2,042
In-scope convention candidates	1,965
Balanced main sample (20 × 5 repositories)	100

Characteristic	Eligible (%)	Sample (%)
Two largest repositories	95.9	40.0
Custom labels	2.2	13.0
Pointer fields	68.6	38.0
`omitempty` present	77.8	58.0

Table 2: Confirmatory task-selection flow. Every filter is deterministic; the final sample is balanced by repository and deliberately enriches rarer labels and types. It is a stress-test estimand, not a prevalence-weighted sample of the 1,965 candidates.

The paired views unbundle three questions. Local versus raw measures the value of repository evidence. Card versus Profile tests recommendation against description. The two raw-matched comparisons test whether a structured surface helps after holding the source pool and input budget nearly constant. No condition changes the model, label, task order, output schema, or retrieval scoring.

4 Experimental design

4.1 Pilot, main split, and matched follow-up

An 84-task pilot across seven repositories and two Flash model families established the parser, representations, and batching protocol. It also revealed that nearby unmasked JSON tags can make the local condition too informative. We therefore freeze a disjoint main set of 100 sites—20 from each of five repositories—whose 11-line local window contains no other JSON tag. Tasks have multiword or custom names and cover snake, lower-camel, and custom labels. This sparse-local split directly measures how repository evidence fills an otherwise unresolved choice. Table 2 makes the estimand explicit: the headline effect describes convention-ambiguous, sparse-local decisions rather than a prevalence-weighted average over all JSON fields. Concretely, a candidate is pilot-disjoint; has distinct snake and lower-camel transforms; carries an observed snake, lower-camel, or custom label; has zero other JSON tags in its local window; and passes the fixed off-scope identifier filter released in the builder. The five repositories each have at least 20 such candidates. A seeded shuffle first retains one example from each non-empty label class per repository, then fills each repository to 20. The resulting 100 sites span 56 files and 82 structs.

The initial main matrix freezes Local, Choice Card, Choice Profile, Card + raw, and its length-matched Raw evidence control. After the raw condition emerges as strongest, we run a targeted representation follow-up on the same frozen tasks: Profile + raw versus an independently rendered, 0.7%-matched Raw evidence view. We label this pair as a follow-up and use it to test the mechanism suggested by the primary result; the primary raw-versus-local claim comes from the initial matrix.

4.2 Models, protocol, and accounting

We evaluate DeepSeek V4 Flash, GLM 5.3 Flash, and GPT-5.6 Luna through OpenCode 1.18.7. Each model answers every task under every condition, producing $100 \times 3 \times 7 = 2,100$ main and follow-up predictions. Conditions run in isolated sessions and batch ten cases with an identical JSON output contract. Prompts prohibit tool use and sessions expose an intentionally empty working directory. Raw runner streams, parsed answers, session identifiers, input sizes, latency, tokens, and provider-reported costs are retained. We use provider-default decoding; the APIs expose no user-set seed, so repeated evaluations use new isolated sessions rather than claiming seed control. Three

Stage / status	Scores	Cost	Role in the evidence chain
Pilot (exploratory)	672	$0.37	Fix parser, views, batching; retain natural-neighbor subset
Primary + follow-up (frozen/labeled)	2,100	$1.29	Seven paired views; Raw–Local is primary
Robustness suite (pre-registered)	2,880	$2.47	Matched foreign, repeats, one-target calls
No-copy + frontier (pre-registered)	400	$3.23	Remove direct paths; stronger model on locked prefix
Sensitivity audits (zero-call)	0	$0.00	Family/source/auxiliary strata, reweighting, matching error
Matched foreign (pre-registered)	300	$0.21	Closest-score control for the no-copy construction
Post-release holdout (pre-reg. + exploratory)	224	$0.14	Four new repositories; no-copy test, then direct-path diagnostic

Table 3: Experiment timeline. Costs include retained attempts; 70 frontier predictions outside the locked prefix remain costed and exploratory.

malformed primary batches are repaired by splitting the identical cases into two smaller calls. Pilot and confirmation together cost $1.66. All interval estimates are conditional on these fixed model families and provider-default decoding; repeated sessions are inference replicates, not independently sampled model populations.

The artifact and benchmark card pin source commits and licenses; release tasks, prompts, raw streams, scoring code, derived tables, figure generators, and checksums; and documents a complete rebuild through PDF compilation and artifact tests. The review artifact is available at `https://anonymous.4open.science/r/belong/`. Rebuilding statistics requires no API call.

We report semantic choice accuracy and its exact naming and presence components. For each comparison, we compute a model-level difference per task, average the three model differences, and bootstrap the 100 task clusters 10,000 times. This keeps the task, not the 300 correlated model–task rows, as the uncertainty unit. We additionally report directional transitions: a *rescue* changes an incorrect baseline prediction to a correct treatment prediction; a *regression* does the reverse. A dependence sensitivity groups tasks by repository, field, exact label, and presence policy, then gives each of the resulting 78 families equal bootstrap weight. A provenance sensitivity separates files with an explicit generator marker in their first 15 lines from files without one; we call the latter stratum handwritten as an operational label, while recognizing that missing markers can hide generated files. The classification is fixed from source text and released per task.

4.3 Staged evaluation and pre-registered controls

The robustness suite first replaces relevant lines with equal-length foreign excerpts, then repeats Local and three evidence views in fresh sessions, and finally evaluates 30 outcome-blind tasks with one target per call. Exclusive Raw subsequently removes every target-file row, exact target-field row, and identifier mention while preserving the scorer, tasks, models, and character counts. The frontier control evaluates Local and Raw with medium-effort GPT-5.6 Sol on the pre-analysis-locked first five batches (50 tasks); all non-prefix calls remain outside that estimate.

Exclusive Foreign completes the no-copy comparison. It draws only from other repositories, applies the same three exclusions, greedily matches the Exclusive Raw retriever-score profile using visible field identifiers, types, and packages, and truncates to the identical character count. Selection never uses labels or outcomes; the 100-task audit verifies every exclusion and length match. This comparison estimates the effect of the complete same-repository evidence construction—repository source pool plus its closest available retrieved precedents—not repository identity in isolation.

The post-release holdout adds 28 tasks spanning 22 files and 27 structs in four repositories absent from every earlier stage and created after the dated DeepSeek identifier. Its pre-registered matrix compares Local, Exclusive Foreign, and Exclusive Raw with two models. After that no-copy comparison is null, a labeled exploratory condition restores Original Raw to separate concrete reconstruction from cross-file convention abstraction. All 168 pre-registered and 56 exploratory predictions are retained. The complete study contains 6,576 scored predictions and costs $7.706, below the frozen $8 hard stop.

Context view	DeepSeek	GLM	Luna	Mean	Strict mean
Local	56	59	65	60.0	47.7
Choice Card	84	79	80	81.0	73.7
Choice Profile	84	88	89	87.0	78.3
Card + raw	84	88	84	85.3	82.7
Raw evidence	92	95	93	93.3	88.7
Profile + raw	90	94	92	92.0	88.7
Raw evidence (matched)	**96**	**97**	**94**	**95.7**	**93.7**

Table 4: Semantic choice accuracy (%) on 100 main tasks, plus the stricter mean that also requires exact output formatting. Every task is answered under every view.

	All ($n = 100$)			Aux-free ($n = 80$)			Handwritten ($n = 60$)		
Condition	Name	Presence	Joint	Name	Presence	Joint	Name	Presence	Joint
Local	87.0	70.0	60.0	83.8	74.2	61.7	93.9	77.8	73.3
Exclusive Foreign	86.0	73.3	62.3	82.5	77.5	63.7	89.4	82.8	73.3
Exclusive Raw	92.3	89.3	81.7	90.4	90.4	80.8	95.0	89.4	84.4
Original Raw	97.3	95.3	93.3	96.7	94.6	92.1	98.9	93.9	92.8

Table 5: Semantic component accuracy (%). Both name and presence improve under same-repository evidence after removing all same-site auxiliary-label exposures.

4.4 Baselines and audit slices

Four non-LLM baselines predict from the field name and census with the target removed: fixed snake plus global presence (28%), fixed camel plus global presence (22%), repository-majority style plus type-conditioned presence (69%), and transfer from the nearest precedent (73%). The latter two are strong symbolic references but remain below all three models with matched raw evidence. We audit results by repository, target style, field type, availability of an exact-name precedent, and availability of a precedent outside the target file.

5 Results

5.1 Concrete precedents reconstruct local ambiguity

Table 4 gives the central result. The initial raw repository view improves mean accuracy from 60.0% to 93.3%, a 33.3-point task-paired gain [26.3, 40.7]. The effect is large in every model: +36 points for DeepSeek, +36 for GLM, and +28 for Luna. The surface-matched follow-up reaches 95.7%, showing that the result persists under a matched context allocation. Original Raw contains an exact-field precedent for 76 tasks and a same-file precedent for 82. Distinguishing direct retrieval from broader transfer, Exclusive Raw removes both on every task and still reaches 81.7% versus 60.0%, a +21.7-point gain [15.0, 28.7]. Reweighting the five repository effects to all 1,965 eligible candidates retains +32.6 points [23.6, 41.6].

The auxiliary-label audit localizes rather than weakens this result. Exactly 20 target declarations contain the gold JSON name in a non-JSON tag: all are `form` tags, 19 are from Stripe and one from gRPC-Gateway, and all are generated/schema-derived. Table 5 reports semantic components on all tasks, the 80 auxiliary-label-free tasks, and the marker-negative handwritten stratum. On auxiliary-label-free tasks, Original Raw over Local gains +12.9 name [7.1, 20.0], +20.4 presence [13.8, 27.1], and +30.4 joint points [22.5, 38.3]. Exclusive Raw over Exclusive Foreign retains +7.9 [2.9, 13.3], +12.9 [7.5, 18.8], and +17.1 [10.0, 24.6]. Thus relevant evidence improves both name and presence when no auxiliary tag exposes the name.

The transition matrix shows a strongly directional gain. Figure 2 counts all 300 paired model–task decisions. Raw evidence rescues 107 local errors and regresses only seven, while 173 decisions remain correct and 13 remain wrong. Strict output-protocol accuracy moves from 47.7% to 88.7%; the semantic metric's smaller but still decisive gain persists under normalization of redundant `,omitempty` suffixes.

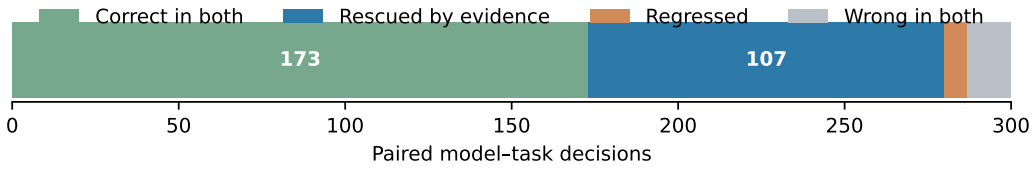

Figure 2: Paired transitions from Local to the initial Raw evidence view across three models. Concrete precedents rescue 107 decisions and regress seven.

5.2 No-copy evidence survives matched controls

The equal-length foreign-context control probes the central mechanism. Foreign code raises accuracy only from 60.0% to 61.3% (+1.3 points; 95% CI [–6.0, 8.3]), while relevant repository lines reach 93.3%. At identical evidence-block length, relevance therefore contributes 32.0 points [24.7, 39.7]. The contrast is positive in every model: +29 points for DeepSeek, +39 for GLM, and +28 for Luna.

The closest-available no-copy comparison tightens this evidence. Exclusive Foreign reaches 62.3%, only +2.3 points over Local [-4.3, 8.7]; Exclusive Raw reaches 81.7%, a +19.3-point advantage [13.0, 25.7], with positive per-model gaps of +15, +20, and +23. Matching is approximate, yet lower/upper-half effects remain +24.7 [14.7, 34.7] and +14.0 [6.0, 22.0]. Within the 60 marker-negative handwritten tasks, Exclusive Raw reaches 84.4% versus 73.3% for Exclusive Foreign: +11.1 points [3.3, 18.9].

The result also repeats rather than depending on one realization. Runs 1–3 produce gains of 33.3, 37.7, and 36.3 points; Raw accuracy has only 1.0 point run-level standard deviation. One-target-per-call evaluation reaches 87.8% versus 58.9%, a 28.9-point paired gain [14.4, 43.3]. Its aggregate differs by only 3.3 points from the same subset's ten-target calls, leaving batching unable to account for the effect. The per-model point estimate is positive throughout, although DeepSeek's +16.7-point unbatched interval [–3.3, 36.7] includes zero.

Direct-path removal and frontier capability expand the boundary. Exclusive Raw reaches 81.7%, a 21.7-point gain [15.0, 28.7] over Local; all three models improve by 20–23 points, and the closest-available control above supports same-repository relevance. Medium-effort GPT-5.6 Sol also leaves a large gap: Raw reaches 100.0% versus 80.0% Local on the frozen 50-task prefix, +20.0 points [10.0, 32.0], with ten rescues and no regressions. Thus neither direct copying nor a stronger model closes the need for repository evidence. Table 6 and Figure 3 expose the complete control sequence. All 70 over-dispatched predictions remain costed and labeled exploratory in the artifact; they do not alter the locked confirmatory estimate.

The effect becomes smaller but remains positive outside the sparse-local stress design. Before constructing the main split, the disjoint pilot sampled seven repositories without filtering neighboring tags. Among its 64 tasks that retain 1–10 natural JSON tags, Raw reaches 93.8% versus 83.6% Local: +10.2 points [3.1, 16.4] across two models. We report this frozen-output analysis as exploratory external-validity evidence, not as part of the confirmatory headline. Reweighting its seven repository effects to the broader convention-candidate population gives +10.5 points.

A time-separated holdout sharpens that external-validity result. The 28 tasks come from four licensed repositories created after the dated DeepSeek identifier, span 22 files and 27 structs, contain at most two neighboring JSON tags, and expose no same-name auxiliary label. Original Raw reaches 78.6% versus 62.5% Local: +16.1 points [1.8, 32.1], with 11 rescues and 2 regressions; both models improve. The pre-registered no-copy contrast is instead -3.6 points [-16.1, 8.9]. Concrete reconstruction therefore transfers on this four-repository holdout. The null no-copy contrast localizes the transferred signal to concrete precedents.

5.3 Transfer and mechanism boundaries

Raw–Local is positive in every repository. Task-bootstrap point estimates and 95% intervals are Fiber +13.3 [6.7, 20.0], go-github +26.7 [8.3, 43.3], gRPC-Gateway +63.3 [46.7, 78.3], Kubernetes API

Comparison	Base	Added view	Δ	95% CI
Run 1: Local	60.0	93.3	+33.3	[26.3, 40.7]
Foreign vs Local	60.0	61.3	+1.3	[–6.0, 8.3]
Relevant vs foreign	61.3	93.3	+32.0	[24.7, 39.7]
Filtered foreign vs Local	60.0	62.3	+2.3	[–4.3, 8.7]
Exclusive vs filtered	62.3	81.7	+19.3	[13.0, 25.7]
Handwritten filtered pair	73.3	84.4	+11.1	[3.3, 18.9]
One target per call	58.9	87.8	+28.9	[14.4, 43.3]
Full no-copy retrieval	60.0	81.7	+21.7	[15.0, 28.7]
Frontier model (50)	80.0	100.0	+20.0	[10.0, 32.0]
Pilot: natural neighbors	83.6	93.8	+10.2	[3.1, 16.4]
Post-release Original Raw†	62.5	78.6	+16.1	[1.8, 32.1]
Post-release no-copy‡	62.5	58.9	–3.6	[–16.1, 8.9]
Match MAE: Q1=.095; median=.204; Q3=.403; max=1.065.				

Table 6: Robustness controls (accuracy and paired changes in percentage points). Both foreign/no-copy controls use 100 tasks and three models; the handwritten pair uses 60, unbatched uses 30, and the frontier control uses 50. The disjoint pre-main pilot row uses 64 tasks with 1–10 visible neighboring tags, seven repositories, and two models. Post-release rows use 28 tasks, four new repositories, and two models; ‡ is pre-registered and † is its labeled exploratory mechanism follow-up. Intervals resample task clusters.

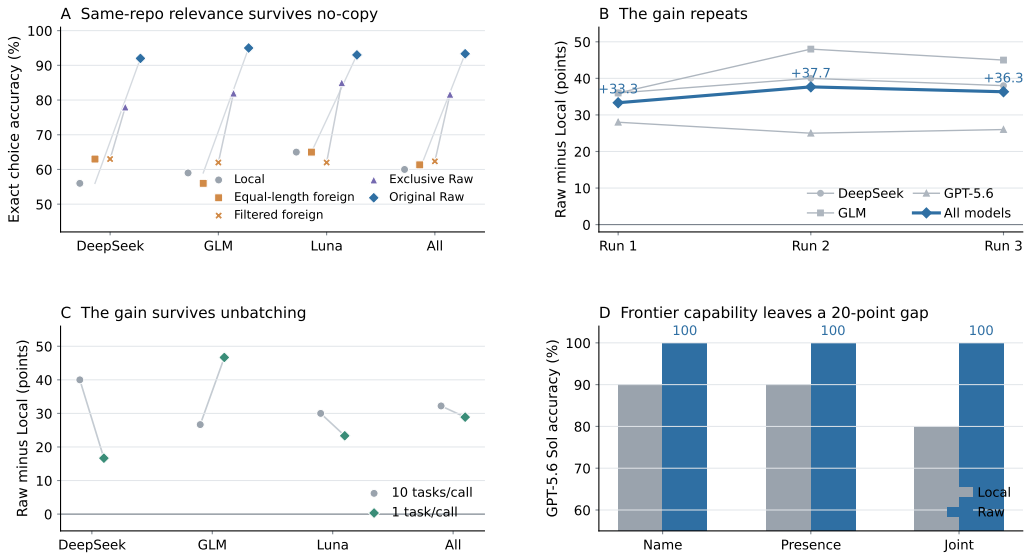

Figure 3: Five interventions preserve the main result. **A**: both foreign controls stay near Local, while same-repository evidence excluding every direct target path retains a large gain. **B**: all three fresh-session realizations are positive. **C**: one-target calls preserve the aggregate effect. **D**: frontier-model name, presence, and joint accuracy all rise with evidence.

+18.3 [8.3, 28.3], and Stripe +45.0 [30.0, 60.0]. Descriptive Exclusive–Local point estimates are likewise positive in all five repositories; their task intervals (three excluding zero) are released in the artifact. Resampling the five repository-level Raw effects gives a 33.3-point gain [18.0, 50.7]; leaving out any one repository retains 25.8–38.3 points. Reweighting those effects to the repository frequencies of all 1,965 eligible candidates yields +32.6 points [23.6, 41.6], close to the balanced estimate. Exclusive Raw is positive in every repository point estimate, with a repository-bootstrap interval of [10.3, 35.0].

Table 7 shows that Main-set Original Raw remains positive with and without an exact-field precedent, whereas post-release gain concentrates where one is retrieved. New-repository effects are +21.4, +7.1,

Population	Exact	n	Local → Raw	Δ [95% CI]
Main	Present	76	54.4 → 92.1	+37.7 [28.9, 46.5]
Main	Absent	24	77.8 → 97.2	+19.4 [11.1, 27.8]
Post-release	Present	13	73.1 → 100.0	+26.9 [7.7, 50.0]
Post-release	Absent	15	53.3 → 60.0	+6.7 [–10.0, 26.7]

Table 7: Original-Raw reconstruction by exact-field-precedent availability. The new holdout localizes transfer; the larger main set also retains a no-exact-row effect.

+0.0, and +35.7; leave-one-out effects span +9.5–21.4. Task bootstrap describes site uncertainty in this fixed holdout; four-cluster sensitivities diagnose dominance, not repository-population uncertainty.

The effect also survives two dependence checks. Giving each of 78 exact field–label families equal weight retains +28.0 points [20.7, 35.3] for Raw and +18.3 points [11.8, 25.1] for Exclusive Raw. The source-marker audit classifies 60 tasks as handwritten and 40 as generated or schema-derived. Raw improves the handwritten stratum by 19.4 points [12.2, 26.7], while Exclusive Raw retains +11.1 points [3.9, 18.3]. The generated stratum has a larger, not exclusive, effect; moreover, the matched handwritten comparison retains +11.1 points [3.3, 18.9]. These are operational source strata: a file without a header marker can still be generated.

6 Related work

Repository-level methods retrieve cross-file context for generation, while CrossCodeEval and RepoBench measure cross-file completion (Shrivastava et al., 2023; Zhang et al., 2023; Wu et al., 2024; Ding et al., 2023; Liu et al., 2024b). BELONG instead fixes generation to one exact developer label and intervenes on evidence, estimating paired choice reconstruction rather than completion quality. Completion scores can conflate convention inference with copying an answer already present in retrieved context. BELONG makes that frontier measurable by reporting exact-field availability, removing direct paths, and matching the remaining evidence against foreign repositories.

SWE-bench and SWE-agent ground models in repositories and executable environments, but their pass/fail outcomes allow multiple convention choices (Jimenez et al., 2024; Yang et al., 2024). BELONG measures this orthogonal adaptation capability directly before full patch evaluation.

Retrieval conditions predictions on inspectable records; long inputs can bury relevant evidence, motivating compression (Lewis et al., 2020; Liu et al., 2024a; Jiang et al., 2024). Earlier convention and cache models learn project regularities for names, properties, and next tokens (Hindle et al., 2012; Allamanis et al., 2014; Tu et al., 2014; Hellendoorn & Devanbu, 2017; Raychev et al., 2015). BELONG changes the target to one natural repository-specific choice among valid implementations and changes only supplied evidence. Its foreign, provenance, and no-copy controls distinguish conditional reconstruction from generic examples, generated boilerplate, and direct precedents. Error wrapping, constructor selection, protobuf options, and logging fields can reuse the benchmark's mining, masking, retrieval, interventions, and transition scoring; parsers, label equivalence, and source filters remain task-specific.

7 Demonstrated Scope and Extensions

BELONG establishes exact repository-conditioned choice reconstruction for Go JSON fields across five main repositories and three models. Its paired intervention measures response to supplied evidence, while the four-repository holdout demonstrates transfer from concrete precedents and localizes the null no-copy result. The dated DeepSeek comparison provides the holdout's temporal guarantee. End-to-end patch acceptance, exact foreign matching, unbounded convention frontiers, stronger generated-code markers, and broader post-release samples define the next tests of abstraction without concrete precedents.

8 Conclusion

Across 6,576 predictions, Original Raw adds 33.3 points; after direct-path removal, same-repository evidence still beats matched foreign by 19.3 [13.0, 25.7]. Four new repositories delimit the scope: concrete precedents add 16.1 points, whereas no-copy abstraction does not transfer.

AI Use Statement

Generative AI tools assisted with research ideation and execution, retrieval and discovery of related work, drafting portions of the manuscript, and language polishing. The authors reviewed and verified all AI-assisted outputs, experimental results, citations, code, and claims, and take full responsibility for the final content.

References

Miltiadis Allamanis, Earl T. Barr, Christian Bird, and Charles Sutton. Learning natural coding conventions. In *Proceedings of the 22nd ACM SIGSOFT International Symposium on Foundations of Software Engineering*, 2014.

Yangruibo Ding, Zijian Wang, Wasi Uddin Ahmad, Hantian Ding, Ming Tan, Nihal Jain, Murali Krishna Ramanathan, Ramesh Nallapati, Parminder Bhatia, Dan Roth, and Bing Xiang. CrossCodeEval: A diverse and multilingual benchmark for cross-file code completion. In *Advances in Neural Information Processing Systems*, 2023.

Vincent J. Hellendoorn and Premkumar Devanbu. Are deep neural networks the best choice for modeling source code? In *Proceedings of the 2017 11th Joint Meeting on Foundations of Software Engineering*, 2017. doi: 10.1145/3106237.3106290.

Abram Hindle, Earl T. Barr, Zhendong Gabel, Zhendong Su, and Premkumar Devanbu. On the naturalness of software. In *Proceedings of the 34th International Conference on Software Engineering*, 2012.

Huiqiang Jiang, Qianhui Wu, Chin-Yew Lin, Yuqing Yang, and Lili Qiu. LongLLMLingua: Accelerating and enhancing LLMs in long context scenarios via prompt compression. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics*, 2024.

Carlos E. Jimenez, John Yang, Alexander Wettig, Shunyu Yao, Kexin Pei, Ofir Press, and Karthik Narasimhan. SWE-bench: Can language models resolve real-world GitHub issues? In *International Conference on Learning Representations*, 2024.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Kuttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. Retrieval-augmented generation for knowledge-intensive NLP tasks. In *Advances in Neural Information Processing Systems*, 2020.

Nelson F. Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. Lost in the middle: How language models use long contexts. *Transactions of the Association for Computational Linguistics*, 12:157–173, 2024a.

Tianyang Liu, Canwen Xu, and Julian McAuley. RepoBench: Benchmarking repository-level code auto-completion systems. In *International Conference on Learning Representations*, 2024b.

Veselin Raychev, Martin Vechev, and Andreas Krause. Predicting program properties from "big code". In *Proceedings of the 42nd Annual ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages*, 2015. doi: 10.1145/2676726.2677009.

Disha Shrivastava, Hugo Larochelle, and Daniel Tarlow. Repository-level prompt generation for large language models of code. In *Proceedings of the 40th International Conference on Machine Learning*, 2023.

Zhaopeng Tu, Zhendong Su, and Premkumar Devanbu. On the localness of software. In *Proceedings of the 22nd ACM SIGSOFT International Symposium on Foundations of Software Engineering*, 2014. doi: 10.1145/2635868.2635875.

Di Wu, Wasi Ahmad Zhang, Wenhao Zhang, Yao Lin, Hanting Chen, Mike Wang, Dian Yu, Yasheng Wang, Rui Zhang, et al. Repoformer: Selective retrieval for repository-level code completion. In *Proceedings of the 41st International Conference on Machine Learning*, 2024.

John Yang, Carlos E. Jimenez, Alexander Wettig, Kilian Lieret, Shunyu Yao, Karthik Narasimhan, and Ofir Press. SWE-agent: Agent-computer interfaces enable automated software engineering. In *Advances in Neural Information Processing Systems*, 2024.

Fengji Zhang, Bei Chen, Yue Zhang, Jacky Keung, Jin Liu, Daoguang Zan, Yi Mao, Jian-Guang Lou, and Weizhu Chen. RepoCoder: Repository-level code completion through iterative retrieval and generation. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, 2023.