RIGHT VALUES, WRONG RECORDS: BENCHMARKING AND IMPROVING RECORD FORMATION IN PROMPT-TO-DATA SYSTEMS

Anonymous authors
Paper under double-blind review

ABSTRACT

Prompt-to-data systems can recover correct values but assign them to wrong records without shared source identifiers. End-to-end evaluation combines this decision with extraction and structural generation. **RecordBench** isolates **record formation** using controlled source-table candidates and document extractions: 2,382 episodes from 807 tables, including explicit-identity controls. **Reference-Guided Assembly (RGA)** uses a frozen tabular foundation model to infer candidate compatibility from reference records in context, then forms records by constrained global assignment, preserving observed values under complete or partial matching. Component-macro target-cell accuracy reaches **76.81%** on controlled and **68.94%** on extracted candidates; controlled RGA exceeds Qwen3.5-27B's executed tools by **18.23 percentage points**. Same-backbone scoring ablations and separate assignment interventions establish the roles of reference-conditioned predictive distributions, cross-target conditioning and global competition in RGA.

1 INTRODUCTION

A prompt-to-data (P2D) system must recover facts and associate them correctly. Attaching a production measurement to the wrong facility can distort site comparisons and database updates, even when every extracted value and data type is valid (Figure 1). **Right values do not guarantee right records.** We study this **record-formation** problem when reports lack shared identifiers.

EVAPORATE (Arora et al., 2023) extracts structured views from documents; DocETL (Shankar et al., 2025) and LOTUS (Patel et al., 2025) provide document-processing pipelines and semantic operators; TEXT2DB (Jiao et al., 2024) integrates extractions into databases. An end-to-end output score alone cannot distinguish missing facts from correct facts assigned to wrong entities. These failures require different remedies: better extraction or better association. Evaluating formation therefore requires fixing available facts while testing their record membership.

RecordBench provides this evaluation through 2,382 episodes from 807 tables, including explicit-identity controls. Controlled candidates preserve source values to isolate assignment; shared document extractions retain upstream errors. Identity controls distinguish association from identification. Within each setting, methods share references and candidates, enabling comparisons across P2D operators, language pathways and learned baselines.

Reference-Guided Assembly (RGA) uses attribute relationships in separate reference rows from the same source. Measurements compatible with particular facility properties, and with one another, help resolve ambiguous associations. A frozen tabular foundation model scores observed anchor–bundle combinations through reference-conditioned predictive distributions. Constrained global assignment resolves competition for candidates, preserving observed values under complete or partial matching.

Experiments on RecordBench demonstrate that RGA improves record formation with both controlled and extracted candidates, outperforming Qwen3.5-27B's tool-execution pathway by **18.23 percentage points** on controlled candidates (Table 2). Same-backbone ablations and assignment interventions establish the roles of reference-conditioned predictive distributions, cross-target conditioning, and global competition. Additional Text2DB experiments evaluate RGA within database-update workflows involving partial assignment.

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.

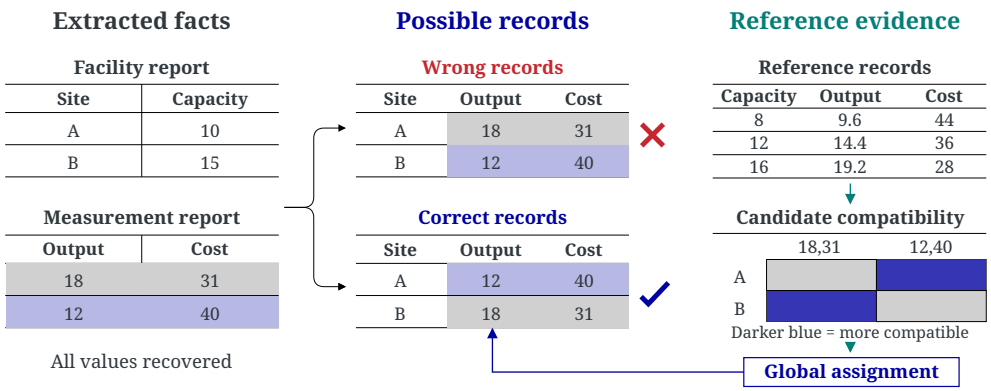


Figure 1: **Right values can form wrong records.** The same extracted facts yield different record memberships; reference compatibility guides global formation. Darker blue indicates higher compatibility; value shading is unchanged across tables. The example, references and matrix are illustrative.

2 BENCHMARK: RECORDBENCH FOR EVALUATING RECORD FORMATION

2.1 TASK DEFINITION

An episode supplies documents D, schema S, reference records R and candidate inventory $\mathcal{C}$: n anchors $X = \{x_i\}_{i=1}^{n}$ and m observed target bundles $Y = \{y_j\}_{j=1}^{m}$. The schema defines record granularity, anchor attributes and numeric targets. Record formation assigns bundles to anchors under source-defined constraints: completely for a closed inventory, or partially when facts are missing, candidates are extra or no suitable match exists. Bundles may have missing target cells.

RecordBench constructs independently ordered reports from TabLib (Eggert et al., 2023) records, retaining attributes/values but omitting shared cross-report keys for statistical association; source association supplies the target. References/queries use distinct original rows of the same source table, and extracted-setting references share the query cohort. This shared-source association task gives all methods the same references; component grouping and duplicate checks are specified in Appendix S1. Standard closed episodes use 128 references, eight anchors, eight bundles, two numeric targets and up to 24 anchor features, isolating placement from unmatched decisions.

2.2 TWO CANDIDATE SETTINGS

Controlled candidates (Table 1) preserve source values, isolating record formation under a fixed inventory. **Extracted candidates** share one DocETL extraction, retaining omissions and incorrect values to test whether assignment improves imperfect document-derived tables.

Table 1: **RecordBench settings use disjoint source tables.** Another 256 identity controls share controlled sources: 2,382 total episodes, 807 tables, 733 components.

Setting	Episodes	Tables	Components	Candidate origin and role
Controlled candidates	768	128	128	Source-preserved values; isolate record formation
Extracted candidates	1,358	679	605	Shared DocETL output; include extraction errors

Controlled statistical-association L and explicit-identity K use the first 128 components in the fixed locked-test ordering of a 2,000-table, 16,000-episode bank. H is constructed separately using a shared

DocETL extraction pipeline. Each interface is scored separately. Appendix S1 specifies screening, components, targets and manifests.

2.3 Evaluation

Target-cell accuracy credits source-correct values per anchor and target column; **complete-record accuracy** requires all anchor attributes and both targets correct. Equivalent numeric values receive equal credit. Assignment validity is separate: one-to-one structure does not imply correctness. Scores average seeds within episodes, episodes within components and components equally; paired 95% intervals use 2,000 component-bootstrap resamples. The appendix details metrics, partitions, failures, identity controls and secondary evaluations.

Uniform random bijections yield 21.75% target-cell accuracy on L and 23.44% on H. Gold-informed maximum-weight assignments under identical inventories, metrics and one-to-one constraints permit cell/record accuracy of 100%/100% on controlled candidates and 90.71%/70.81% on H (Table S3). These metric-specific ceilings distinguish unavailable facts from scoring/assignment loss, not observational identifiability.

2.4 Connection to database integration

Database updates attach facts to existing/new entities, allowing unequal counts, missing targets and unmatched anchors. Our separate reference-eligible Text2DB cohort uses 9 development tasks/6 components and 23 test tasks/22 components, with one, two or three Qwen3.5-4B extraction runs. Existing related rows supply references, excluding query-entity overlaps. Component-macro changed-cell F1 compares initial/final states compatibly with Text2DB; task-macro and association F1 complement it. Methods share each task/run-subset's saved facts and constraints. This measures final database outcomes, versus RecordBench's direct formation test (eligibility, support and materialization: Appendix S9).

3 Reference-Guided Assembly

RGA uses a frozen tabular foundation model (TFM), conditioned in context on reference records, to score observed anchor–bundle compatibility and form records through constrained global assignment (Figure 2).

3.1 Compatibility from reference records

Let p_θ be a frozen predictor conditioned on R, with targets standardized by reference statistics. For observed coordinates O_j and their order π, bundle compatibility is

$$q_\pi(y_{j,O_j} \mid x_i, R) = \prod_{t=1}^{|O_j|} p_\theta(y_{j,\pi_t} \mid x_i, y_{j,\pi_{<t}}, R)\,.$$

Each factor predicts a target from the anchor and preceding targets, observed in references and supplied by the scored candidate at the query: proposed records are evaluated without generating replacements. Following variable-order density ensembles (Uria et al., 2014), RGA averages forward/reverse factorizations into costs

$$C_{ij} = -\log\left[\tfrac{1}{2}q_{\rightarrow}(y_{j,O_j} \mid x_i, R) + \tfrac{1}{2}q_{\leftarrow}(y_{j,O_j} \mid x_i, R)\right].$$

Orders coincide for one observed target; otherwise conditioning lets complementary targets affect compatibility. Reference rows require available labels for scored coordinates; missing candidate cells remain unfilled. RGA uses **LimiX-2**'s context-conditioned modeling (Zhang et al., 2026). Its **mean-residual** and **marginal-density** ablations respectively compare point predictions and independently score targets, holding backbone, references, inventories, preprocessing and global solver fixed. TabPFN-3.5 (Jäger et al., 2026) provides external baselines and a separate within-backbone mean/marginal comparison.

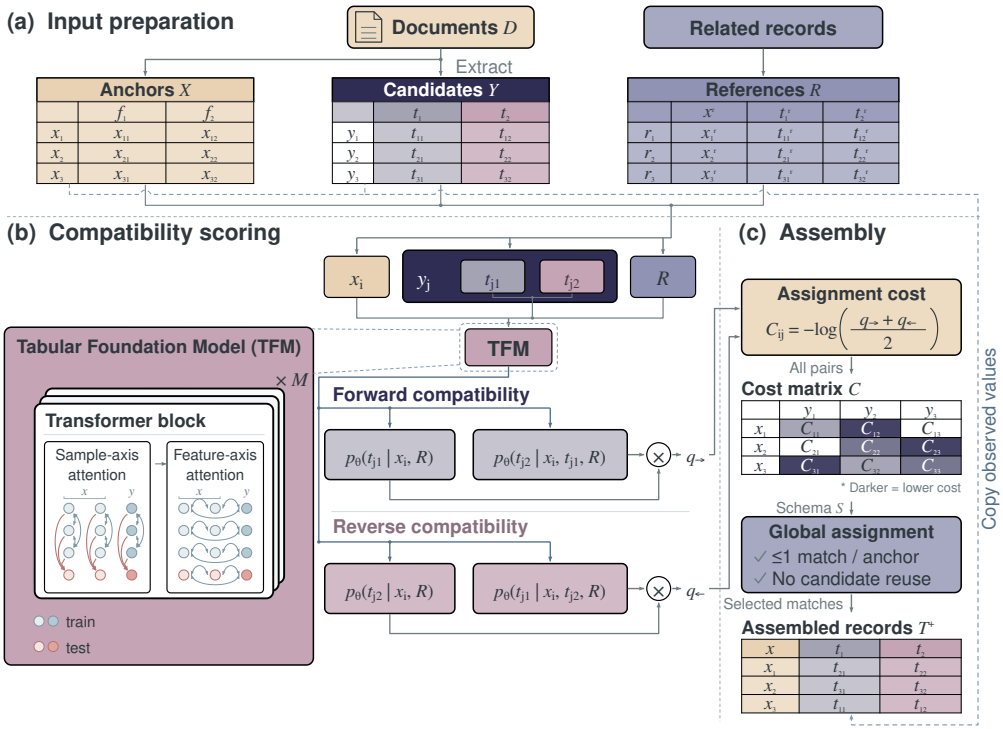


Figure 2: **RGA scores and jointly assembles records.** (a) Extracted anchors/bundles. (b) Frozen TFM (primarily LimiX-2): bidirectional compatibility from separate-row references. (c) Higher compatibility lowers cost; assignment copies selected values along the dashed path. Closed assignment is shown; the same score supports unmatched anchors (Section 3.2).

3.2 Global record formation

To resolve independent matches competing for the same candidate, RGA solves

$$(A^*, u^*) = \underset{(A,u)\in\mathcal{F}(S)}{\arg\min} \left[\sum_{i,j} A_{ij} C_{ij} + \lambda \sum_i u_i \right], \qquad T^+ = \text{materialize}(X, Y, A^*).$$

Binary A and unmatched indicators u_i satisfy $\sum_j A_{ij} + u_i = 1$, $\sum_i A_{ij} \le 1$ and fixed/forbidden source matches. Closed RecordBench assignment has $n = m$, $u_i = 0$; partial assignment adds dummies with cost λ, selected on development tasks (Appendix S9) and fixed for evaluation. Both use identical scores and a linear-assignment solver (Crouse, 2016; Virtanen et al., 2020).

Materialization copies selected observed values, leaving unmatched targets unresolved without generating facts or rerunning extraction. **Record review** ranks records in an initial assignment π_0 by assigned-edge cost $C_{i,\pi_0(i)}$ for inspection, rather than reconstructing it.

4 Experiments

Paired comparisons fix references and source-component support. LimiX-2 is primary throughout; TabPFN-3.5 independently tests distributional scoring.

4.1 Existing systems reveal a record-formation gap

Comparators. Extracted-inventory **P2D operators** are DocETL-Join, LOTUS-Join (native semantic joins) and LOTUS-Global (complete assignments). **Language pathways** use Qwen3.5-

4B/27B (Qwen Team, 2026) and Llama-3.1-8B-Instruct (Grattafiori et al., 2024) for direct assignment, matrix generation, reference-trained tools or one-cycle direct-output refinement (Madaan et al., 2023). **Conventional learners** use regression or matched/contrasted reference pairs before the common solver, including CatBoost (Prokhorenkova et al., 2018) and ExtraTrees (Geurts et al., 2006). **External TFMs** TabICLv2, TabPFN-3 and TabPFN-3.5 supply point/distributional costs. Table 2 compares external baselines with RGA; Table 5 isolates LimiX-2 scoring. Comparisons fix references/candidates (decoding, interfaces, failures: Appendix S2).

Table 2: **Complete RGA versus external baselines: 23 configurations.** Cell/record accuracies are percentages (↑). Controlled: 768 episodes/128 components; extracted: 1,358/605. Qwen3.5-4B uses one seed; Qwen3.5-27B/Llama use three. Dashes: unmeasured; failures retain cohort weight. LimiX-2 ablations: Table 5; paired RGA intervals: Table S5.

Method / configuration	Controlled cell	Controlled record	Extracted cell	Extracted record
P2D operators				
DocETL-Join	–	–	25.18	13.87
LOTUS-Join	–	–	3.09	1.57
LOTUS-Global	–	–	28.95	17.95
Language-model pathways				
Qwen3.5-4B: direct	35.37	29.20	35.28	21.98
Qwen3.5-4B: generated matrix	21.57	15.10	23.13	11.63
Qwen3.5-4B: executed tools	57.35	52.28	51.48	37.03
Qwen3.5-4B: self-refinement	34.39	28.48	–	–
Qwen3.5-27B: direct	44.68	38.87	–	–
Qwen3.5-27B: generated matrix	31.36	24.92	–	–
Qwen3.5-27B: executed tools	58.58	53.28	–	–
Qwen3.5-27B: self-refinement	44.13	39.02	–	–
Llama-3.1-8B: direct	24.51	18.09	–	–
Llama-3.1-8B: generated matrix	21.30	14.83	–	–
Llama-3.1-8B: executed tools	57.29	51.80	–	–
Llama-3.1-8B: self-refinement	16.94	12.41	–	–
Conventional learners + assignment				
Regression-CatBoost	58.22	52.85	51.49	37.05
Pair-CatBoost	64.53	59.37	57.13	42.05
ExtraTrees	59.97	54.88	52.61	38.56
External tabular-model baselines				
TabICLv2: mean residual	64.44	59.44	56.27	41.78
TabPFN-3: mean residual	66.16	61.31	57.56	42.47
TabPFN-3.5: mean residual	68.32	63.74	60.12	45.01
TabPFN-3.5: marginal density	75.39	71.31	68.37	51.57
Proposed complete method				
RGA with LimiX-2	**76.81**	**72.85**	**68.94**	**52.37**

Capability gap. Controlled direct/matrix results (Table 2) show the importance of compatibility estimation in the evaluated pathways. Reference-trained tools improve Qwen3.5-4B and Llama; Qwen3.5-27B tools and Pair-CatBoost improve further, leaving substantial room. Against TabPFN-3.5 marginal, RGA has paired gains of 1.42 points [-0.07, 2.87] controlled and 0.57 [-0.19, 1.32] extracted; these comparisons do not establish a significant difference.

Identity versus compatibility. Shared keys yield direct accuracy of 96.04% for Qwen and 91.27% for Llama, versus 76.32% for unchanged statistical RGA (Table S6). Reliable keys call for key-aware identification; statistical association tests references without direct identity.

4.2 RGA IMPROVES EXTRACTED P2D OUTPUTS

On 1,358 DocETL-extracted episodes, RGA improves Qwen's executed-tool cell accuracy by 17.46 points [15.74, 19.16] (Table 3); record accuracy rises from 37.03% to 52.37% (Table 2). Thus, gains reach final structured outputs with extracted facts unchanged.

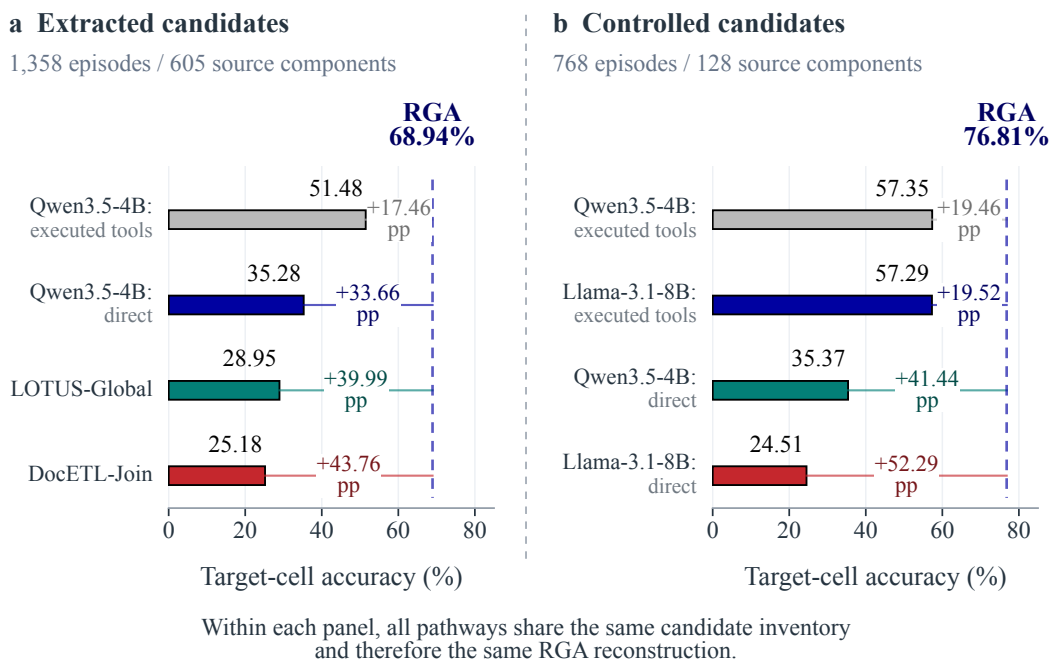


Figure 3: **Different pathways, shared facts, one RGA reconstruction.** Horizontal bars rank baseline accuracy within panels; dashed blue lines show common RGA accuracy from the shared candidate bank. (a) DocETL: 1,358 episodes/605 components. (b) Controlled: 768 episodes/128 components. Gains are percentage points; paired intervals: Tables 3 and S7.

Table 3: **Improvement on shared DocETL extractions.** Base/+RGA: cell accuracy (%); gains: points with paired 95% intervals. DocETL-Join and LOTUS-Global are evaluated formation operators. The shared inventory yields one RGA reconstruction.

Base record-formation pathway	Base	+RGA	Gain [95% interval]
DocETL-Join	25.18	**68.94**	+43.76 [41.29, 46.06]
LOTUS-Global	28.95	**68.94**	+39.99 [37.45, 42.46]
Qwen3.5-4B: direct	35.28	**68.94**	+33.66 [31.37, 35.98]
Qwen3.5-4B: executed tools	51.48	**68.94**	+17.46 [15.74, 19.16]

DocETL-Join and LOTUS-Global also improve (Table 3; Figure 3), showing a post-extraction use of reference relationships: placing recovered facts together without extracting them again.

4.3 RECORD FORMATION WITHIN DATABASE INTEGRATION

Text2DB development/test components are disjoint. Three saved Qwen3.5-4B runs supply inventories: unions merge normalized anchor identities and bundle values, retaining maximum within-run multiplicity. RGA, its two scoring variants and two external baselines share references and each task/run-subset inventory. Per method/budget, development calibration selects the largest unmatched cost with mean paired loss from the optimum within one standard error, then fixes it for evaluation; all budgets select the same method-specific thresholds (Appendix S9). RGA and its marginal-density

variant share a checkpoint; its mean-score variant uses a separate point-prediction checkpoint. These are final-update/partial-assignment comparisons; RecordBench ablations isolate same-backbone scoring.

Table 4: **Text2DB F1 (%) by extraction budget.** Component (C)/Association (A) F1 weight 22 components equally; Task (T) F1 weights 23 tasks equally. Budgets one/two average all three runs/pair unions; three uses the full union. Each budget shares inventories across RGA, its two scoring variants and two external baselines, without score-based run/pair selection. Thresholds/failures: Appendix S9.

	1 extraction			**2 extractions**			**3 extractions**		
Method	C	T	A	C	T	A	C	T	A
RGA (LimiX-2)	**18.0658**	**17.2804**	**25.7576**	**24.7181**	**23.6434**	**27.5758**	**25.3674**	**24.2645**	**29.0909**
RGA: marginal	**18.0658**	**17.2804**	**25.7576**	**24.7181**	**23.6434**	**27.5758**	**25.3674**	**24.2645**	**29.0909**
RGA: mean	17.9767	17.1951	**25.7576**	24.6290	23.5581	**27.5758**	25.2783	24.1793	**29.0909**
TabPFN	17.9767	17.1951	**25.7576**	24.6290	23.5581	**27.5758**	25.2783	24.1793	**29.0909**
Pair-CatBoost	17.9767	17.1951	25.2525	24.4774	23.4132	26.9697	25.1268	24.0343	28.4848

Table 4's component-macro changed-cell F1 rises with extraction budget across methods, showing candidate availability's role in database quality. Its marginal-density variant attains the same F1 on this cohort; Table 5 isolates scoring components on RecordBench.

4.4 RGA REDUCES THE GAP WITH CORRECT CANDIDATE VALUES

On 768 controlled episodes, RGA exceeds Qwen's executed tools by 19.46 points [16.41, 22.65] and Pair-CatBoost by 12.28 [9.32, 15.38] (Table 2). Pair-CatBoost learns from matched/contrasted reference pairs: gains therefore concern reference use with candidate values and support fixed.

Among 745 perfectly extracted episodes/419 components, RGA reaches 79.41%, versus 59.41% for tools and 66.52% for Pair-CatBoost. This diagnostic complements the full extracted cohort (Table 2): a formation gap persists even with every candidate recovered correctly.

4.5 THE IMPROVEMENT EXTENDS ACROSS THE EVALUATED MODELS

Language families and scale. Against Llama's three-seed outputs, the shared RGA reconstruction reaches 76.81%, gaining 19.52 points [16.45, 22.59] over tools versus Qwen's 19.46 (Table 2; matrices/refinement: Table S6). Qwen3.5-27B improves direct/tools on the same 128 components across three complete seeds, yet RGA gains 18.23 points [15.19, 21.27] over tools. Shared keys yield 96.88% (Table S14). The formation gap persists across the evaluated model sizes and pathways.

Frozen tabular backbones. TabPFN-3.5 marginal improves over mean scoring in both settings with backbone, references, preprocessing and solver fixed (Table 2; Figure S1). Together with LimiX-2, this supports candidate likelihood across backbones.

4.6 SAME-BACKBONE ABLATIONS AND ASSIGNMENT INTERVENTIONS

Table 5 ablates RGA with backbone, references, inventory and global solver fixed. Mean residual replaces distributional compatibility with point discrepancies; marginal density removes cross-target conditioning.

Distributional compatibility. Marginal density improves over mean residual by 6.42/7.28 points on controlled/extracted candidates. Complete RGA adds 0.80 [0.03, 1.62]/0.79 [0.29, 1.30] over marginal and 7.22 [5.14, 9.43]/8.08 [6.90, 9.25] over mean residual.

Reference correspondence. On one fixed episode per controlled component (128 total), intact references yield 77.39%. Permuting bundles across reference anchors preserves values but reduces accuracy to 22.31%; reordering complete rows preserves relationships and yields 77.00%. Attribute co-occurrence, not inventory alone, supplies the signal.

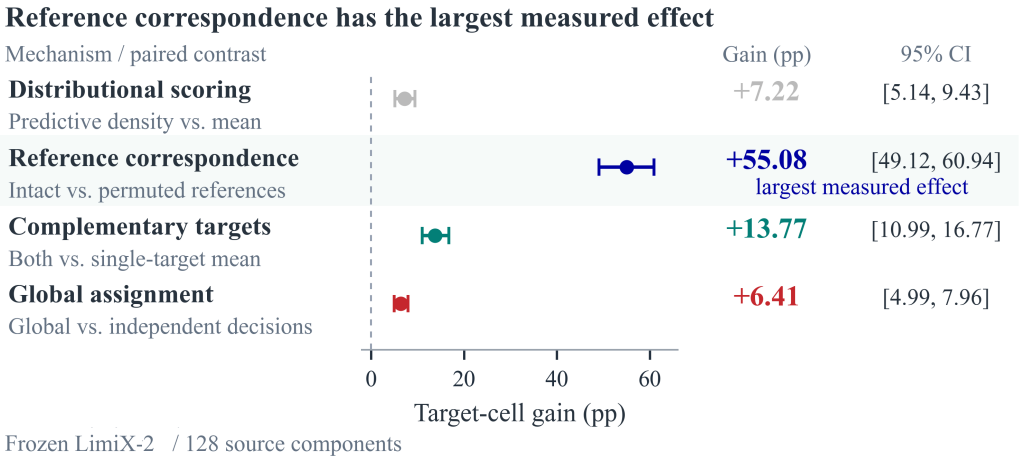


Figure 4: **Reference-conditioned compatibility and global competition in RGA.** Points/bars: paired cell gains/95% component-bootstrap intervals; colors: mechanisms. All fix LimiX-2 and average 128 components. Scoring/assignment use all 768 controlled episodes; correspondence/targets use one fixed episode per component. Interventions are separate, not additive.

Table 5: **Same-backbone RGA scoring ablations** (Section 3.1): component-macro cell/record accuracy (%) on full controlled/extracted cohorts, with frozen LimiX-2 and the same global solver.

RGA configuration	Controlled cell	Controlled record	Extracted cell	Extracted record
Mean-residual variant	69.59	65.20	60.87	45.85
Marginal-density variant	76.01	72.05	68.15	51.73
Complete RGA	**76.81**	**72.85**	**68.94**	**52.37**

Complementary targets. On that cohort, both targets yield 77.39% versus the predetermined single-target-alternative mean of 63.62%: +13.77 points [10.99, 16.77], scoring every alternative on both final columns. Complementary observed attributes help identify compatible records.

Global competition. With costs fixed, global assignment exceeds independent row-wise selection by 6.41 points [4.99, 7.96] on controlled and 5.21 [4.45, 5.96] on extracted candidates, resolving conflicts among plausible matches.

4.7 Reference availability and candidate competition

More references raise accuracy from 66.80% to 84.38% (Figure 5a). With 128 candidates, RGA retains an 11.72-point advantage over the same-model mean control on fixed focal queries (Figure 5b). Both grids fix 16 sources, isolating evidence/competition from population changes.

Assigned-record compatibility also supports review without re-extraction: at a 25% budget, error precision is 89.35% versus random review's 69.92% on common eligible controlled outputs (Figure S3).

5 Related Work

Document-to-structured-data systems. DocETL (Shankar et al., 2025), LOTUS (Patel et al., 2025), EVAPORATE (Arora et al., 2023) and Palimpzest (Liu et al., 2025) support extraction/semantic processing; TEXT2DB (Jiao et al., 2024), DTBench (Guo et al., 2026) and Doc2DB-Bench (Liang et al., 2026) evaluate integration/extraction/database construction. RecordBench complements them by isolating formation.

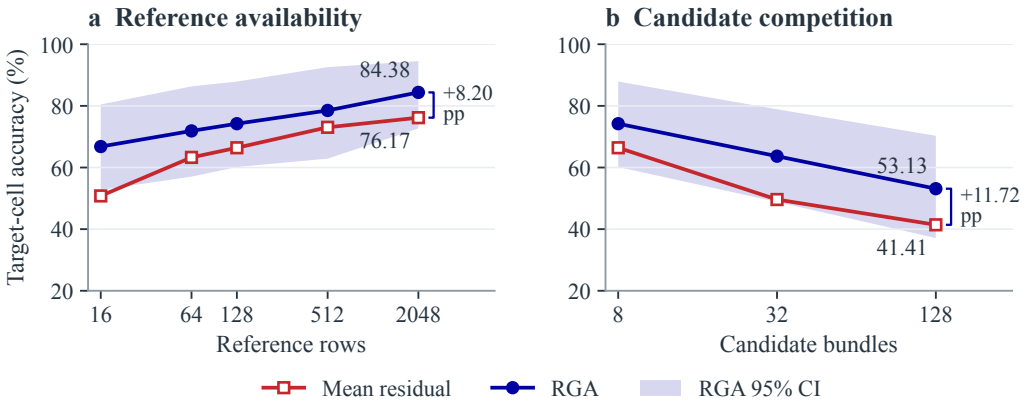


Figure 5: **More references help; RGA retains its advantage under competition.** (a) Nested references. (b) Growing pools for eight fixed focal queries. Both fix LimiX-2 and 16 sources. Pale blue: RGA 95% component-bootstrap intervals. Final Δ: RGA minus mean-residual accuracy at the largest size. Full curves: Figure S2.

Table learning and conditional prediction. TabICLv2 (Qu et al., 2026), TabPFN-3 (Grinsztajn et al., 2026) and LimiX-2 (Zhang et al., 2026) predict in context from references; uLEAD-TabPFN (Lu et al., 2026) uses conditional dependencies for anomaly detection. RGA scores candidate assignments with reference-conditioned distributions and resolves them jointly.

Record linkage and unknown correspondences. Probabilistic linkage models links, nonlinks and unresolved cases (Fellegi & Sunter, 1969; Jaro, 1989); Bayesian bipartite linkage jointly infers matches, optionally leaving uncertain matches unresolved (Sadinle, 2017). Shuffled regression studies regression/permutation recovery with unknown correspondences (Pananjady et al., 2018). RGA contributes a reference-conditioned formation interface, cross-field compatibility learned from separate reference rows and evaluation on fixed extracted inventories.

Data repair and table evaluation. Swapping repair reassigns within-tuple misplaced values (Sun et al., 2020); HoloClean (Rekatsinas et al., 2017) combines statistics/constraints. TabXEval (Pancholi et al., 2025), TabReX (Anvekar et al., 2026) and TabStruct (Jiang et al., 2026) assess table quality/structural fidelity. RecordBench directly tests source-defined membership in fixed/observed inventories.

6 Discussion

RGA makes record formation a modular step between extraction and database materialization, using reference-conditioned compatibility to assemble observed values. Its scores rely on reference records capturing relationships relevant to the query cohort; robustness under reference–query distribution shift remains to be evaluated. Current experiments focus on numeric target bundles under closed or partial one-to-one assignment. Extending this formulation to mixed-type targets and richer relational constraints is a direction for future work.

7 Conclusion

RecordBench separates formation from value recovery, revealing gaps across P2D operators, language pathways and learned association methods even with correct candidates. RGA reduces these gaps on controlled and extracted candidates through reference-conditioned compatibility and closed/partial global assignment. The benchmark and baselines operationalize the importance of both recovered facts and their record membership.

AI USE STATEMENT

Generative AI tools assisted research framing, experiment design, implementation, testing, literature discovery, data preparation, interpretation, manuscript drafting and editing, and plotting-code development. Numerical results and quantitative figures derive from saved experimental records. A conceptual illustration was generated with AI assistance and checked against its stated example; the method overview and quantitative plots were rendered from editable code. The authors retain responsibility for the claims, citations, data, code and final text.

ETHICS STATEMENT

The benchmark transforms source-table records into controlled reports. Use of source-derived content remains subject to original source permissions. RGA reconstructs observed candidate values rather than generating replacements. Applications to sensitive records require legitimate access and domain oversight.

REPRODUCIBILITY STATEMENT

The experiment artifacts retain fixed task manifests, per-episode outputs, evaluation code, source-component aggregation and figure source data. The appendix specifies interfaces, model versions, failure handling and uncertainty estimates. Model weights and source payloads are obtained under their respective access terms.

REFERENCES

N. S. Altman (1992). An Introduction to Kernel and Nearest-Neighbor Nonparametric Regression. The American Statistician, 46(3), pp. 175–185.

Tejas Anvekar; Junha Park; Aparna Garimella; Vivek Gupta (2026). TabReX: Tabular Reference-less eXplainable Evaluation. Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pp. 47015–47030.

Simran Arora; Brandon Yang; Sabri Eyuboglu; Avanika Narayan; Andrew Hojel; Immanuel Trummer; Christopher Ré (2023). Language Models Enable Simple Systems for Generating Structured Views of Heterogeneous Data Lakes. Proceedings of the VLDB Endowment, 17(2), pp. 92–105.

David F. Crouse (2016). On implementing 2D rectangular assignment algorithms. IEEE Transactions on Aerospace and Electronic Systems, 52(4), pp. 1679–1696.

Gus Eggert; Kevin Huo; Mike Biven; Justin Waugh (2023). TabLib: A Dataset of 627M Tables with Context. arXiv:2310.07875.

Ivan P. Fellegi; Alan B. Sunter (1969). A Theory for Record Linkage. Journal of the American Statistical Association, 64(328), pp. 1183–1210.

Pierre Geurts; Damien Ernst; Louis Wehenkel (2006). Extremely randomized trees. Machine Learning, 63(1), pp. 3–42.

Aaron Grattafiori et al. (2024). The Llama 3 Herd of Models. arXiv:2407.21783v3.

Léo Grinsztajn; Klemens Flöge; Oscar Key; Felix Birkel; Philipp Jund; Brendan Roof; Mihir Manium; Shi Bin Hoo; Magnus Bühler; Anurag Garg; Dominik Safaric; Jake Robertson; Benjamin Jäger; Simone Alessi; Adrian Hayler; Vladyslav Moroshan; Lennart Purucker; Philipp Singer; Alan Arazi; Julien Siems; Jan Hendrik Metzen; Georg Grab; Nick Erickson; Siyuan Guo; Eliott Kalfon; Simon Bing; David Salinas; Clara Cornu; Lilly Charlotte Wehrhahn; Diana Kriuchkova; Kursat Kaya; Lydia Sidhoum; Marie Salmon; Jerry Chen; Madelon Hulsebos; Yann LeCun; Samuel Müller; Bernhard Schölkopf; Sauraj Gambhir; Noah Hollmann; Frank Hutter (2026). TabPFN-3: Technical Report. arXiv:2605.13986v2.

Yuxiang Guo; Zhuoran Du; Nan Tang; Kezheng Tang; Congcong Ge; Yunjun Gao (2026). DT-Bench: A Synthetic Benchmark for Document-to-Table Extraction. Proceedings of the 32nd ACM SIGKDD Conference on Knowledge Discovery and Data Mining V.2, pp. 8942–8953.

Trevor Hastie; Robert Tibshirani; Jerome Friedman (2009). The Elements of Statistical Learning: Data Mining, Inference, and Prediction. Second edition. Springer.

Arthur E. Hoerl; Robert W. Kennard (1970). Ridge Regression: Biased Estimation for Nonorthogonal Problems. Technometrics, 12(1), pp. 55–67.

Benjamin Jäger; Nick Erickson; Léo Grinsztajn; Felix Birkel; Klemens Flöge; Oscar Key; Kürşat Kaya; Jonas Kübler; Adèle Frankel; Tobias Schröder; Anurag Garg; Jan Hendrik Metzen; David Salinas; Simon Bing; Kristina Collins; Tuana Çelik; Vahid Balazadeh; Lydia Sidhoum; Tomás Pereda; Brendan Roof; Andrej Tschalzev; Siyuan Guo; Philipp Singer; Lennart Purucker; Jake Robertson; Marie Salmon; Philipp Jund; Jerry Chen; Diana Kriuchkova; Arthur Cahu; Eliott Kalfon; Adrian Hayler; Georg Grab; Vitor Monteiro; Lilly Wehrhahn; Dominik Safaric; Clara Cornu; Alan Arazi; Rylee Grace; Simone Alessi; Mihir Manium; Bernhard Schölkopf; Yann LeCun; Madelon Hulsebos; Sauraj Gambhir; Noah Hollmann; Frank Hutter (2026). TabPFN-3.5: Technical Report. arXiv:2609.17895v2.

Matthew A. Jaro (1989). Advances in Record-Linkage Methodology as Applied to Matching the 1985 Census of Tampa, Florida. Journal of the American Statistical Association, 84(406), pp. 414–420.

Xiangjian Jiang; Nikola Simidjievski; Mateja Jamnik (2026). TabStruct: Measuring Structural Fidelity of Tabular Data. International Conference on Learning Representations.

Yizhu Jiao; Sha Li; Sizhe Zhou; Heng Ji; Jiawei Han (2024). Text2DB: Integration-Aware Information Extraction with Large Language Model Agents. Findings of the Association for Computational Linguistics: ACL 2024, pp. 185–205.

Zhuowen Liang; Zhengxuan Zhang; Jiayang Wang; Jiazhuo Chen; Nan Tang (2026). Beyond Tables: Doc2DB-Bench for Relationally Faithful Document-to-Database Construction. arXiv:2608.08459.

Chunwei Liu; Matthew Russo; Michael Cafarella; Lei Cao; Peter Baile Chen; Zui Chen; Michael Franklin; Tim Kraska; Samuel Madden; Rana Shahout; Gerardo Vitagliano (2025). Palimpzest: Optimizing AI-Powered Analytics with Declarative Query Processing. 15th Conference on Innovative Data Systems Research (CIDR).

Sha Lu; Jixue Liu; Stefan Peters; Thuc Duy Le; Craig Xie; Lin Liu; Jiuyong Li (2026). uLEAD-TabPFN: Uncertainty-aware Dependency-based Anomaly Detection with TabPFN. arXiv:2604.20255.

Aman Madaan; Niket Tandon; Prakhar Gupta; Skyler Hallinan; Luyu Gao; Sarah Wiegreffe; Uri Alon; Nouha Dziri; Shrimai Prabhumoye; Yiming Yang; Shashank Gupta; Bodhisattwa Prasad Majumder; Katherine Hermann; Sean Welleck; Amir Yazdanbakhsh; Peter Clark (2023). Self-Refine: Iterative Refinement with Self-Feedback. Advances in Neural Information Processing Systems, 36, pp. 46534–46594.

Ashwin Pananjady; Martin J. Wainwright; Thomas A. Courtade (2018). Linear Regression With Shuffled Data: Statistical and Computational Limits of Permutation Recovery. IEEE Transactions on Information Theory, 64(5), pp. 3286–3300.

Vihang Pancholi; Jainit Sushil Bafna; Tejas Anvekar; Manish Shrivastava; Vivek Gupta (2025). TabXEval: Why this is a Bad Table? An eXhaustive Rubric for Table Evaluation. Findings of the Association for Computational Linguistics: ACL 2025, pp. 22913–22934.

Liana Patel; Siddharth Jha; Melissa Pan; Harshit Gupta; Parth Asawa; Carlos Guestrin; Matei Zaharia (2025). Semantic Operators and Their Optimization: Enabling LLM-Based Data Processing with Accuracy Guarantees in LOTUS. Proceedings of the VLDB Endowment, 18(11), pp. 4171–4184.

Liudmila Prokhorenkova; Gleb Gusev; Aleksandr Vorobev; Anna Veronika Dorogush; Andrey Gulin (2018). CatBoost: unbiased boosting with categorical features. Advances in Neural Information Processing Systems, 31.

Jingang Qu; David Holzmüller; Gaël Varoquaux; Marine Le Morvan (2026). TabICLv2: A Better, Faster, Scalable, and Open Tabular Foundation Model. International Conference on Machine Learning.

Qwen Team (2026). Qwen3.5: Towards Native Multimodal Agents. Official release, February 2026.

Theodoros Rekatsinas; Xu Chu; Ihab F. Ilyas; Christopher Ré (2017). HoloClean: Holistic Data Repairs with Probabilistic Inference. Proceedings of the VLDB Endowment, 10(11), pp. 1190–1201.

Mauricio Sadinle (2017). Bayesian Estimation of Bipartite Matchings for Record Linkage. Journal of the American Statistical Association, 112(518), pp. 600–612.

Shreya Shankar; Tristan Chambers; Tarak Shah; Aditya G. Parameswaran; Eugene Wu (2025). DocETL: Agentic Query Rewriting and Evaluation for Complex Document Processing. Proceedings of the VLDB Endowment, 18(9), pp. 3035–3048.

Yu Sun; Shaoxu Song; Chen Wang; Jianmin Wang (2020). Swapping Repair for Misplaced Attribute Values. 2020 IEEE 36th International Conference on Data Engineering (ICDE), pp. 721–732.

Benigno Uria; Iain Murray; Hugo Larochelle (2014). A Deep and Tractable Density Estimator. Proceedings of the 31st International Conference on Machine Learning, PMLR 32(1), pp. 467–475.

Pauli Virtanen; Ralf Gommers; Travis E. Oliphant; Matt Haberland; Tyler Reddy; David Cournapeau; Evgeni Burovski; Pearu Peterson; Warren Weckesser; Jonathan Bright; Stéfan J. van der Walt; Matthew Brett; Joshua Wilson; K. Jarrod Millman; Nikolay Mayorov; Andrew R. J. Nelson; Eric Jones; Robert Kern; Eric Larson; C J Carey; İlhan Polat; Yu Feng; Eric W. Moore; Jake VanderPlas; Denis Laxalde; Josef Perktold; Robert Cimrman; Ian Henriksen; E. A. Quintero; Charles R. Harris; Anne M. Archibald; Antônio H. Ribeiro; Fabian Pedregosa; Paul van Mulbregt; SciPy 1.0 Contributors (2020). SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. Nature Methods, 17(3), pp. 261–272.

Xingxuan Zhang; Gang Ren; Hao Yuan; Hao Zou; Hongze Tan; Hui Wang; Jianhao Song; Jiansheng Li; Jiayao Zhang; Jinghan Zhang; Kaifang Li; Lang Mo; Li Mao; Mingchao Hao; Nuo Xu; Rui Ding; Ruiji Zhang; Shuyang Li; Siyu Mei; Tianyang Zhang; Weiyang Mu; Yancheng Dong; Yongxian Wei; Yuan Xue; Yuanrui Wang; Yue He; Zijia Yang; Ziyun Li; Dongzhe Li; Fuqiang Wang; Jiandong Liu; Jiawei Chen; Jiaxin Du; Kaijie Cheng; Kehan Li; Lei Sun; Linjun Zhou; Ningbo Dai; Qi Wang; Renzhe Xu; Shaoxing Du; Shumeng Yang; Wang Lu; Wenjing Chu; Xiannan Huang; Xiaoyu Lin; Xing Ai; Xinyan Han; Xuanyue Li; Xuanyue Su; Xukun Zhang; Yan Lu; Yaxin Zhang; Yi Qin; Yifei Huang; Yihan Xu; Yongle Lv; Yuanyuan Jiang; Yushan Han; Peng Cui (2026). LimiX-2: A Contextual Mechanism Network Towards General Structured-Data Intelligence. arXiv:2609.17488v1.

SUPPLEMENTARY MATERIAL

S1 BENCHMARK CONSTRUCTION AND EVALUATION

S1.1 CONSTRUCTION FLOW AND FROZEN EVALUATION SUBSETS

RecordBench renders independently ordered anchor and observation reports from processed TabLib tables. Report-local identifiers do not encode the cross-report association in the statistical condition; the evaluator retains the source row relation. Controlled candidates preserve source values, while H uses saved shared DocETL inventories, including incorrect values and extraction failures. Source answers are not supplied to the scorer or language pathways.

The controlled builder first excludes 1,458 previously used table IDs from the sampling frame, then samples 5,000 candidate tables round-robin across row-count and column-count strata with seed 9152026. Profiling retains 4,449 eligible tables: 546 candidates fail with value/eligibility errors and five exceed the 128 MiB source-payload limit. Candidate tables and prior tasks enter one grouping graph. Removing all candidate components connected to prior tasks excludes another 1,031 eligible

tables, leaving 3,418 tables in 2,601 components. Selection in a fixed randomized component order, retaining whole components, produces the bank in Table S1. The 1,458 initial exclusions precede the 5,000-table sample.

Table S1: **Controlled-bank construction and partitions.** The final three rows partition the constructed bank. Constructed episodes are not all evaluated episodes.

Stage / partition	Tables	Components	Episodes
Profiled candidates	5,000	–	–
Eligible candidates	4,449	–	–
After prior-component exclusion	3,418	2,601	–
Constructed bank	2,000	1,646	16,000
Development	390	330	3,120
Validation	380	329	3,040
Locked test	1,230	987	9,840

Components are sorted by minimum table ID and shuffled with the fixed construction random state. Component positions modulo five assign development (0), validation (1) and locked test (2–4); all selected tables in a component remain together. Within locked test, tasks are sorted by `(component_id, task_id)`, retaining the first task per component. The first 128 components form L/K; the next 64 form the separately reported S7 extension. Each table supplies six statistical-association episodes and two explicit-identity controls on different query blocks (Table S2).

Table S2: **Frozen evaluation membership.** The final two columns count episodes by document evidence. K shares L's tables and components, counted once in the main union. H is constructed separately.

Subset	Tables	Components	Statistical	Identity
L: controlled	128	128	768	0
K: identity controls	128	128	0	256
H: extracted	679	605	1,358	0
Main union	807	733	2,126	256
S7 extension	64	64	384	128

H uses all 679 source tables in its separate test split. For each table, the frozen runner sorts task IDs, selects the first target-rotation task and takes the first query block from each of two cohorts, yielding 1,358 episodes. Task and episode IDs independently seed the final candidate permutation. Failed extractions remain in this denominator. H is constructed independently of the 2,000-table controlled bank. The checked H and L/K manifests have no shared table IDs, task IDs, episode keys, source families or normalized schemas.

Development evaluation uses a retained 100-component subset of the 330 constructed development components. The 64-component extension, 16-source operating-regime experiments, repeated seeds and interventions do not enlarge the main union in Table S2. Language pathways in the controlled setting receive documents, references and candidates; H comparisons consume the shared extracted candidates and references.

S1.2 Target columns, features and reference records

Controlled construction (L/K and S7). Each table is sampled to at most 8,192 rows with seed 9152026. Column names are converted to strings, duplicate headers are excluded and duplicate full rows are removed. A column is numeric if at least 95% of its nonmissing values convert to finite numbers. Eligible targets must also have at least 12 distinct values, at least 70% nonmissing

coverage and names outside the identifier pattern (including ID, index, rank, code, ZIP/postal and UUID). Eligible targets are permuted using seed 9152026 plus the integer encoded by the first eight hexadecimal table-ID characters; the first two are selected without consulting evaluated method scores.

After dropping rows missing either target, at least 192 records must remain. Anchor features prioritize long-text columns, then follow original column order, retaining between three and 24 features. Long text has median nonmissing string length at least 40 or 90th-percentile length at least 120. The first 128 records form references; the next 64 form eight disjoint eight-query blocks. Each query contributes an intact two-target candidate bundle; bundles are randomly permuted and retain separate identities even when their values coincide. The first six episodes omit cross-report identity links; the final two include them. Field-ledger and attribute-sentence templates alternate across episodes.

Extracted-candidate construction (H). The builder applies the 95% numerical-convertibility threshold and identifier filter to the catalog's numeric candidates, shuffles with seed 9142026 and selects target pairs by rotation. Each task uses two cohorts with at least 192 target-complete records each. Within a cohort, the first 128 records are references, the next 32 are calibration records and the next 32 are queries; H evaluates the first eight-query block. Each task stores 256 references across cohorts; each episode receives its matching 128-record slice. All 1,338 available extraction inputs contain 128 references; the remaining 20 inventories are missing.

References and queries thus share a source table and, for H, a cohort, while using different original row positions. This is a same-source association setting, not an independently sourced transfer test. The construction does not establish strict equality of their distributions or resolve entities across reference and query records.

S1.3 Components, duplicates and source coverage

A source component is the connected component (transitive closure) of table-level grouping links. Source families are GitHub repositories (owner/repository), or host domains for other sources. Controlled construction links tables sharing a source family, normalized nonidentifier schema or sampled content signature. Schema normalization lowercases column names, removes nonalphanumeric characters and sorts them. Content keys use the selected anchors and targets: JSON-encode the first 256 processed rows, truncate SHA-256 hashes to 20 hexadecimal characters and retain the first eight sorted signatures. Any shared key connects tables before component-level splitting.

H grouping uses source families and normalized schemas. Controlled construction additionally uses content signatures and excludes components connected to previously used tasks. Content signatures are unavailable for H, limiting cross-bank content checks. Shared-source grouping prevents splitting a recorded component across partitions; it does not remove every near-duplicate table. The absence of H–L/K overlap in recorded identifiers, source families and schemas does not establish semantic, entity-level or pretraining disjointness.

Of the 2,000 constructed tables, 1,997 (99.85%) are GitHub-hosted; the other hosts are `tutton.org`, `forum.dothraki.org` and `fantom.gsc.riken.jp`, one table each. H has 678 GitHub tables and one from `bioinformatics.psb.ugent.be`; all 128 L/K and 64 S7 tables are GitHub-hosted. The archive retains repository-level provenance and table-size strata. These counts describe hosting sources; semantic-domain coverage was not annotated.

Full-row deduplication precedes numeric normalization and feature projection, which can make previously different rows equal. The 16,000 constructed episodes contain 140 query-record occurrences equal to references after processing and 65 repeated query-record occurrences. L/K contains nine reference/query equality occurrences (eight in L, one in K), spanning three tables and five episodes, plus one repeated processed query record. S7 has one reference/query equality occurrence in one statistical-association episode from one table. These are record-occurrence counts, not entity counts; all cases remain in the frozen denominators. Entity resolution was not performed, so different row positions cannot certify different entities.

S1.4 PRETRAINING-OVERLAP AUDIT SCOPE

The audit covers recorded table IDs, task IDs, episode keys, source families, schemas and processed-row equalities. Checkpoint-specific training-membership comparisons have not been performed, so pretraining overlap remains unverified. The reproduction archive retains table IDs, source URLs, repositories, schemas, available source locators and checkpoint records to support model-specific membership checks.

S1.5 METRIC SEMANTICS

For each expected anchor, target-cell accuracy checks whether the emitted bundle contains the source-correct value in each target column. An absent or multiply assigned anchor receives no target credit. A uniquely assigned, compatible-width bundle is compared with the source values using decimal-value equality, with string equality otherwise. The missing-value convention treats `None` and the empty string as equivalent. Complete-record accuracy additionally requires every anchor attribute to match.

Assignment validity checks expected anchor identities, existing candidate identities and one-to-one candidate use. Equivalent target bundles receive value-based credit even when their identifiers differ. Candidate-index correctness is retained as a separate diagnostic. Under uniform random assignment, per-cell accuracy has expectation m_{ik}/n, where m_{ik} counts candidates matching the source value for anchor i and target k. Repeated values therefore change the random expectation from $1/n$.

All planned reconstruction episodes remain in the denominator. A recorded generation, task or scoring failure contributes its failed result. For both evaluated TFM backbones, the 131 failed extracted-candidate episodes comprise 20 missing inventories, 110 numeric-parsing failures and one constant-feature case; all remain in the comparisons.

S1.6 AGGREGATION AND UNCERTAINTY

Let m_{ecs} denote the metric for episode e, component c and seed s. Seeds are averaged within an episode, episodes within a component and components equally. A paired comparison takes the difference within each component before averaging. Confidence intervals use 2,000 component-bootstrap resamples with seed 9152026; all episodes and seed-averaged values in a sampled component remain together.

Cell-transition counts describe changes in correctness, not independent statistical replicates. The extracted-candidate cohort has unequal episode counts per component, so pooled transitions and component-macro accuracy summarize different quantities. Review intervals use common eligible assessment support, as specified below. The perfectly extracted stratum supplements rather than replaces the full extracted-candidate evaluation.

S1.7 RANDOM ASSIGNMENT AND FIXED-INVENTORY RECOVERABILITY

Consider an episode with n anchors, n intact candidate bundles c_j and k target coordinates. Let g_i denote anchor i's gold target bundle. Let w_{ij}^c count the coordinates of c_j equal to g_i, and let w_{ij}^r indicate that all targets match and the extracted anchor attributes equal the source anchor. Equality follows the official evaluator, including Decimal-based numerical comparison. For a uniform random bijection, the exact expected accuracies are

$$B_{\text{cell}} = \frac{\sum_{i,j} w_{ij}^c}{n^2 k}, \qquad B_{\text{record}} = \frac{\sum_{i,j} w_{ij}^r}{n^2}.$$

The corresponding gold-informed inventory ceilings are

$$U_{\text{cell}} = \max_{\pi \in \mathfrak{S}_n} \frac{\sum_i w_{i,\pi(i)}^c}{nk}, \qquad U_{\text{record}} = \max_{\pi \in \mathfrak{S}_n} \frac{\sum_i w_{i,\pi(i)}^r}{n}.$$

Each ceiling solves its own maximum-weight bipartite assignment and is checked with the official evaluator. Their optimal permutations need not coincide. Neither computation changes values, splits bundles or reuses candidates. H's 20 missing extraction inventories receive zero for both random

scores and both ceilings; its other 1,338 inventories contain eight candidates. Scores average episodes within components, then components equally.

Table S3: **Random assignment and inventory ceilings.** Accuracies are percentages; support is episodes/components. Random assignment is a uniform bijection, ignoring explicit keys. Gold-informed ceilings optimize cell and record metrics separately on the same inventory and constraints. Failed inventories retain their denominator weight.

Setting	Support	Random		Inventory ceiling	
		Cell	Record	Cell	Record
H: extracted	1,358 / 605	23.44	11.80	90.71	70.81
L: statistical	768 / 128	21.75	15.15	100.00	100.00
K: identity documents	256 / 128	21.25	14.99	100.00	100.00
S7: statistical	384 / 64	21.50	16.74	100.00	100.00
S7: identity documents	128 / 64	20.46	15.11	100.00	100.00

Main-cohort random expectations match the saved results for all 2,382 episodes; both optimal assignments were evaluator-checked for each of the 2,874 available inventories across the main cohort and S7. The random baseline is not generally $1/n$, because different bundles can share target values. H has 136 episodes missing at least one correct gold bundle and 264 with multiple equal-valued candidates for a gold bundle; the corresponding equal-bundle counts are 117 for L and 34 for K. Equal-valued bundles remain separate candidate objects subject to one-to-one use. Accuracy credits values rather than requiring recovery of distinct source identities for value-equivalent candidates.

For either metric, $1-U$ is loss imposed by the fixed inventory and metric, including anchor corruption for complete-record accuracy. The gap $U -$ method score includes scoring, matching and execution losses within that inventory. Statistical nonidentifiability instead concerns whether observations without gold uniquely determine the association. Neither gap nor the duplicate-bundle rate quantifies that information limit. In particular, a 100% inventory ceiling certifies that a correct assignment exists, not that an unkeyed observation uniquely identifies it. The random identity-control rows ignore document keys and do not replace a key-aware baseline.

S2 Method and Baseline Implementation

S2.1 RGA algorithm

Inputs: reference records R, schema S, anchors X, observed target bundles Y, frozen predictor p_θ and optional initial assignment A_0. **Outputs:** the reconstructed table T^+, assignment A^* and optional record-review scores.

1. Validate candidate identities, bundle widths and the schema's assignment constraints.
2. For each target order, append preceding target columns to the reference features.
3. Evaluate each observed target bundle at every query anchor. Candidate values supply preceding-target features at query time.
4. Average forward and reverse factorization compatibilities and take their negative logarithm to obtain the cost matrix C.
5. Solve the linear assignment problem on C. For A_0, also return the costs of its assigned edges and their within-episode ranking.
6. Materialize the selected anchor and target values without replacing their contents.

The solver uses the common SciPy (Virtanen et al., 2020) linear-assignment interface (Crouse, 2016). A dense $n \times n$ matrix requires $O(n^2)$ storage; scoring cost depends on reference size, the predictor and the number of candidate-conditioned queries. The row-wise control selects the minimum-cost candidate independently for each anchor from the identical saved matrix.

S2.2 PRIMARY BACKBONE AND SCORE CONTROLS

All controlled RecordBench analyses use frozen **LimiX-2**. Inference retains its native eight-member preprocessing ensemble, temperature 0.9 and predictive regression distribution with 5,000 bins. Density evaluation preserves the inverse target transformation. A 10^{-6} probability mixture gives finite compatibility costs in zero-mass bins.

The mean-residual control predicts each reference-standardized target from anchor attributes and sums squared candidate discrepancies. Marginal density evaluates each target independently, without including the other target in its features. The primary joint score averages the two target factorizations. Model weights, references, candidates and the assignment solver are held fixed across these score variants.

TabPFN-3.5 uses native text preprocessing, an eight-estimator ensemble and seed 9182026. String features enter its native TF-IDF branch where applicable. The same full predictive distributions provide mean-residual and marginal-density costs, with the same 10^{-6} mixture for zero-mass bins. The measured TabPFN-3.5 density variant is marginal scoring; it is not the bidirectional joint score used by the primary RGA model.

The TabPFN-3.5 evaluation includes all 1,536 episodes in the 192-component confirmation bank and all 1,358 extracted-candidate episodes. The confirmation bank contains the controlled setting, explicit-identity control and separately reported extension. Source manifests match the corresponding LimiX-2 evaluation. The saved TabPFN-3 (Grinsztajn et al., 2026) and TabICLv2 (Qu et al., 2026) comparators retain their original mean-prediction interfaces and version names.

S2.3 LANGUAGE PATHWAYS

The four pathways use the same observed candidates and reference access:

1. **Direct:** emit a complete anchor-to-candidate mapping.
2. **Matrix:** generate all pair costs in the episode context, then use the common assignment solver.
3. **Executed tools:** choose reference-only Ridge, kNN or CatBoost fits, inspect validation results and candidate costs, and select or combine a matrix.
4. **Self-refinement:** review the saved direct output once and revise it once using the same source and reference evidence.

Ridge (Hoerl & Kennard, 1970) uses reference-only regularization selection; kNN (Altman, 1992) uses five distance-weighted neighbors; CatBoost (Prokhorenkova et al., 2018) uses 200 trees of depth six. Three-fold validation is confined to reference rows. Returned information includes validation error, query predictions and candidate costs. The language model can select a learner matrix, average available matrices or provide its own matrix. Query target answers are not returned.

Self-refinement permits at most six concise issues and 200 feedback words about query assignments. Its end-to-end language cost includes the direct call, feedback and revision. A reused initial call remains part of the workflow. All pathways use zero schema retries; Table S4 specifies the evaluated decoding configurations.

Table S4: **Decoding settings for the completed language comparisons.** Qwen3.5-4B uses seed 9152026; Qwen3.5-27B and Llama use seeds 9152026, 9152027 and 9152028.

Model	Context limit	Output budget	Temperature	Output constraint
Qwen3.5-4B	131,072	2,048	0	JSON schema, non-thinking
Qwen3.5-27B	131,072	2,048	0	JSON schema, non-thinking
Llama-3.1-8B-Instruct	131,072	2,048	0	JSON schema

S2.4 Learned and operator baselines

Regression-CatBoost predicts target values before constructing candidate costs. Pair-CatBoost learns compatibility from matched and contrasted reference pairs. Their configurations are selected on the separate development partition and retained for evaluation. ExtraTrees supplies another conventional regression comparator.

DocETL supplies the shared extracted inventory and its recorded join result. LOTUS-Join uses the native semantic join, which can emit many-to-many matches. LOTUS-Global generates a complete episode-level assignment. The native join's score reflects both output cardinality and association accuracy; LOTUS-Global provides its complete-assignment counterpart.

S3 Complete Model Comparisons

Table 2 compares complete RGA with external baselines, including the additional language pathways in Table S6. Table 5 separately reports the same-backbone scoring ablations. Table S13 retains the primary cross-setting point estimates for reference. The following paired intervals distinguish external comparisons from within-backbone ablations.

Table S5: **Paired RGA differences from external baselines and same-backbone ablations.** Entries are target-cell accuracy differences in percentage points with 95% component-bootstrap intervals. Both settings retain their full evaluation cohorts.

Comparator	Controlled gain [95% interval]	Extracted gain [95% interval]
External baselines		
Qwen3.5-4B: direct	41.44 [36.91, 46.33]	33.66 [31.37, 35.98]
Qwen3.5-4B: generated matrix	55.23 [50.12, 60.35]	45.82 [43.36, 48.28]
Qwen3.5-4B: executed tools	19.46 [16.41, 22.65]	17.46 [15.74, 19.16]
Regression-CatBoost	18.58 [15.54, 21.74]	17.45 [15.77, 19.01]
Pair-CatBoost	12.28 [9.32, 15.38]	11.81 [10.25, 13.19]
ExtraTrees	16.83 [13.93, 19.97]	16.34 [14.61, 18.05]
TabICLv2: mean residual	12.36 [9.79, 15.06]	12.68 [11.16, 14.21]
TabPFN-3: mean residual	10.64 [8.17, 13.29]	11.39 [9.90, 12.81]
TabPFN-3.5: mean residual	8.49 [6.21, 10.86]	8.82 [7.56, 10.11]
TabPFN-3.5: marginal density	1.42 [-0.07, 2.87]	0.57 [-0.19, 1.32]
Same-backbone scoring ablations		
LimiX-2: mean residual	7.22 [5.14, 9.43]	8.08 [6.90, 9.25]
LimiX-2: marginal density	0.80 [0.03, 1.62]	0.79 [0.29, 1.30]

With TabPFN-3.5 held fixed, marginal density improves over mean residual by 7.07 points [4.92, 9.09] on controlled candidates and 8.25 points [7.07, 9.54] on extracted candidates. The LimiX-2 RGA minus TabPFN-3.5 marginal-density differences are 1.42 points [-0.07, 2.87] and 0.57 [-0.19, 1.32], respectively. Both backbones benefit from distributional scoring; the between-system intervals include zero.

S3.1 Language families and direct identity

The identity control supplies a shared explicit key. Direct language assignment uses this information effectively. RGA uses the unchanged statistical scorer. The two conditions distinguish direct identification from reference-conditioned record formation.

Recorded task/output failure rates on controlled candidates are 3.13% for direct, matrix and tool pathways in both language families, 6.25% for Qwen self-refinement and 41.93% for Llama self-refinement. For Qwen direct, matrix and tools, all 24 failures per pathway are context-budget overflows. Failures remain in the reported accuracy denominator.

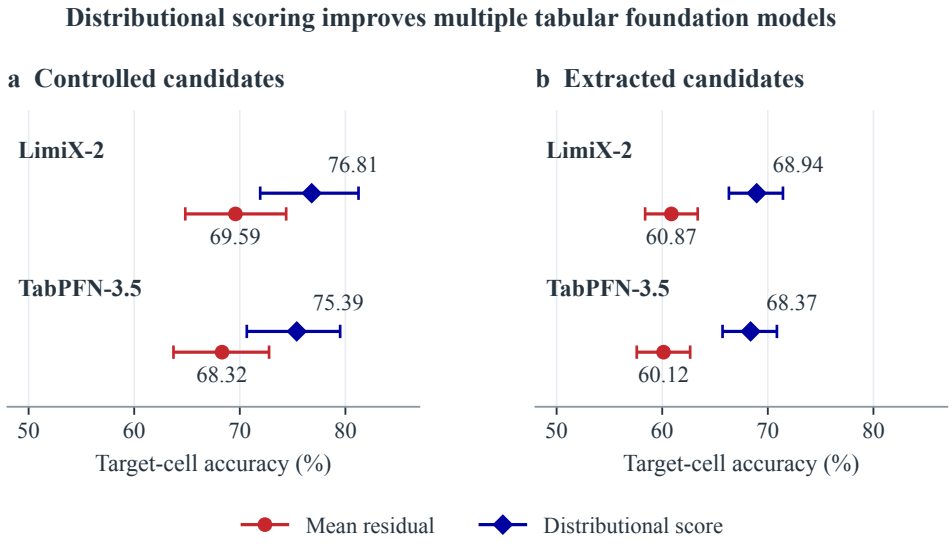


Figure S1: **Distributional scoring across frozen tabular backbones.** Mean-residual scores are compared with marginal density for TabPFN-3.5 and the primary joint score for LimiX-2. Each pair retains the same backbone, references, candidates and global solver. Points and error bars show component-macro accuracy and 95% component-bootstrap intervals.

Table S6: **Qwen3.5-4B and Llama-3.1-8B language pathways.** Accuracy is in percent. Controlled candidates contain 768 episodes; the explicit-identity control contains 256 episodes from the same 128 components. Three Llama seeds are averaged within each episode. The common RGA reconstruction is reported once.

Producer and pathway	Controlled cell	Controlled record	Identity cell	Identity record
Qwen3.5-4B: direct	35.37	29.20	96.04	95.95
Qwen3.5-4B: matrix	21.57	15.10	20.19	13.77
Qwen3.5-4B: tools	57.35	52.28	52.25	46.39
Qwen3.5-4B: self-refinement	34.39	28.48	95.36	95.26
Llama-3.1-8B: direct	24.51	18.09	91.27	90.71
Llama-3.1-8B: matrix	21.30	14.83	19.53	13.04
Llama-3.1-8B: tools	57.29	51.80	56.22	50.85
Llama-3.1-8B: self-refinement	16.94	12.41	56.35	54.93
RGA with LimiX-2	76.81	72.85	76.32	72.61

S4 Post-Hoc Repair and Record Review

S4.1 Fixed-output reconstruction

RGA can improve an existing table without rerunning extraction. It reconstructs the same observed inventory and references for each saved initial output. Different pathways therefore share a final RGA table when their candidate bank is identical. Their corrected/harmed counts differ because the starting records differ.

Under the decoding and one-cycle refinement settings in Appendix S2, self-refinement yields the following outcomes. Qwen self-refinement reaches 34.39% on controlled candidates, compared with 35.37% for the saved direct output and 76.81% for RGA. Relative to direct assignment, self-refinement corrects 780 cells and harms 900; its paired difference is -0.98 points [-2.04, 0.08]. Llama

Table S7: **Paired RGA gains and target-cell transitions.** Gains are percentage points with paired 95% intervals. Controlled rows contain 12,288 target cells per seed; extracted rows contain 21,728. Llama counts are means across three seeds. Extracted-candidate accuracy is component-macro and need not equal the pooled transition fraction.

Initial output	Gain [95% interval]	Corrected	Harmed
Controlled: Qwen direct	41.44 [36.91, 46.33]	5,622.0	530.0
Controlled: Qwen matrix	55.23 [50.12, 60.35]	7,205.0	418.0
Controlled: Qwen tools	19.46 [16.41, 22.65]	2,932.0	541.0
Controlled: Qwen self-refinement	42.42 [37.97, 47.24]	5,759.0	547.0
Controlled: Llama direct	52.29 [47.38, 57.21]	6,861.0	435.0
Controlled: Llama matrix	55.50 [50.43, 60.54]	7,242.7	422.3
Controlled: Llama tools	19.52 [16.45, 22.59]	2,927.3	528.7
Controlled: Llama self-refinement	59.87 [55.01, 64.63]	7,670.7	313.7
Extracted: Qwen direct	33.66 [31.37, 35.98]	8,174.0	932.0
Extracted: Qwen tools	17.46 [15.74, 19.16]	4,805.0	1,040.0

self-refinement corrects 513.7 and harms 1,444.7 cells per seed on average, reaching 16.94%. Full pathway scores remain in Table S6.

S4.2 Review eligibility and budgets

Review takes an existing assignment rather than a new reconstruction. Eligibility requires expected identities, compatible widths, a valid bijection and finite assigned-edge costs. Eligible methods share the same support within a partition. RGA ranks assigned-edge costs from largest to smallest within each episode, breaking ties by stable anchor identifier. At review fraction b, it selects $\lceil bn \rceil$ records, bounded by the available count.

A target-bundle mismatch defines the review error label. Complete-record accuracy separately includes anchor errors. Error-free episodes remain in the precision denominator; recall is averaged on episodes containing errors, with that support recorded. Random-review precision is the expected error prevalence under uniform review. The evaluated budgets are 12.5%, 25% and 50%, with 25% primary.

Table S8: **Error precision at the primary 25% review budget.** All scorers within a setting use common eligible outputs. RGA uses LimiX-2.

Setting	Episodes	Components	RGA precision (%)	Random expectation (%)
Controlled candidates	697	124	89.35	69.92
Extracted candidates	1,093	559	86.97	70.24

S5 Reference and Target Interventions

The intervention cohort contains the first statistical-association episode in source order from each of the 128 controlled-candidate components. Permuting reference target bundles preserves each two-target bundle while changing its reference-anchor correspondence. Whole-row reordering preserves each complete reference record.

Intact minus permuted correspondence yields 55.08 points [49.12, 60.94]. Intact minus reordered rows yields 0.39 [-0.59, 1.51]. Both targets improve on the fixed mean of the single-target alternatives by 13.77 points [10.99, 16.77]. Each single-target method still returns a full assignment evaluated on both final target columns.

Table S9: **Reference correspondence and complementary targets.** All rows use the same LimiX-2 backbone and 128-episode cohort.

Condition	Cell accuracy (%)
Intact references, both targets	77.39
Whole reference rows reordered	77.00
Target bundles permuted across reference anchors	22.31
First target alone	62.50
Second target alone	64.75
Predetermined mean of single-target alternatives	63.62

Global-versus-row-wise comparisons reuse saved compatibility matrices on all 768 controlled and 1,358 extracted episodes. The paired improvements are 6.41 points [4.99, 7.96] and 5.21 [4.45, 5.96]. Only the assignment operation changes.

Ordering probes on the 16-source operating cohort compare assignments after restoring the original identities. Exact repetition preserves assignments on 16/16 sources. Reference-row, query-row and column reordering preserve them on 15/16, 14/16 and 13/16 sources. The evaluator distinguishes changed identifiers from changed assigned values and separately records equivalent-cost assignments.

S6 Reference Availability and Candidate Competition

Reference-size experiments use nested sets of 16, 64, 128, 512 and 2,048 rows. Candidate-size experiments use 8, 32 and 128 candidates at 128 reference rows while tracking the same eight focal queries. Every condition retains the same 16 sources. All 128 source-condition evaluations complete without execution failure.

Table S10: **Reference availability, target-cell accuracy (%).** All score variants use frozen LimiX-2.

Reference rows	Mean residual	Marginal density	RGA
16	50.78	66.41	66.80
64	63.28	73.44	71.88
128	66.41	73.44	74.22
512	73.05	78.52	78.52
2,048	76.17	83.59	84.38

Table S11: **Candidate competition on eight fixed focal queries, accuracy (%).** References contain 128 rows.

Candidates	Mean residual	Marginal density	RGA
8	66.41	73.44	74.22
32	49.61	60.55	63.67
128	41.41	48.83	53.13

At 32 and 128 candidates, whole-query RGA accuracy is 61.43% and 48.73%, respectively. These whole-query scores use different denominators from the fixed-focal comparison. Model weights and scoring remain fixed throughout the grid.

Figure 5 presents the primary scaling comparison in the main paper. Figure S2 adds the marginal-density control.

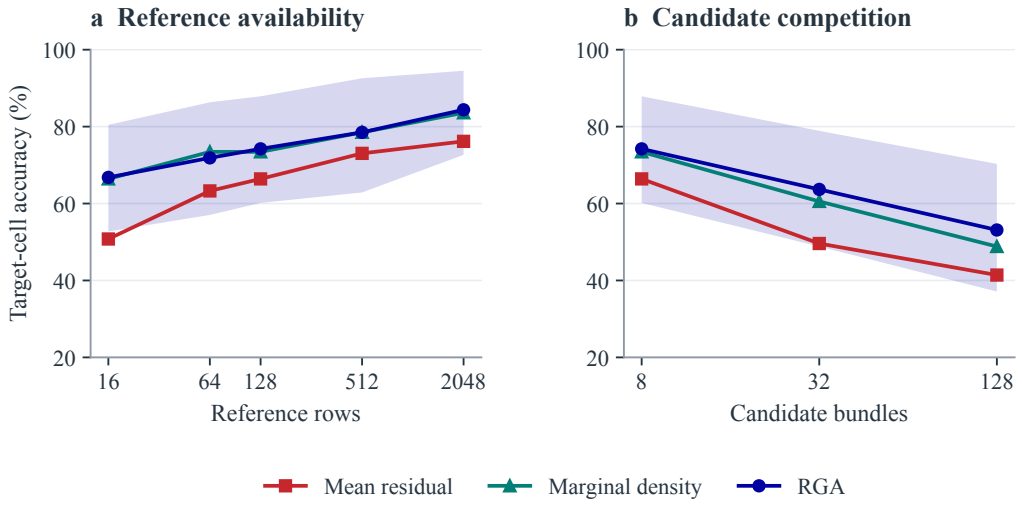


Figure S2: **All score variants on the fixed-source scaling grid.** The same 16-source reference-size and candidate-size experiments as Figure 5, including marginal-density scoring. Shading shows 95% component-bootstrap intervals for RGA. All curves retain the same LimiX-2 model and global assignment solver.

S7 Confirmation Extension and Scope

The confirmation extension contains 64 source tables and 64 components outside the main RecordBench union. It uses components 129–192 in the fixed locked-test ordering after L/K, retaining the first task per component. Its 512 episodes comprise 384 statistical-association episodes and 128 explicit-identity document controls, reported separately in Table S12. Development and extension sources are summarized separately from the main cohort.

On the 384 statistical-association episodes, RGA reaches 78.84% target-cell accuracy and 76.24% complete-record accuracy, compared with 66.59% and 63.83% for the same-model mean-residual control. The paired cell gain is 12.26 points [8.84, 15.85], with 2,000 component-bootstrap resamples and seed 9152026.

Table S12: **The 64-component extension by document evidence.** Entries are cell / complete-record accuracy (%), with episodes averaged within each component and components weighted equally. The three saved LimiX-2 scoring arms have no failed episodes in either stratum.

Evidence stratum	Episodes	Mean	Marginal	RGA
Statistical association	384	66.59 / 63.83	77.90 / 75.36	78.84 / 76.24
Identity documents	128	68.46 / 65.04	78.17 / 74.90	79.59 / 76.17

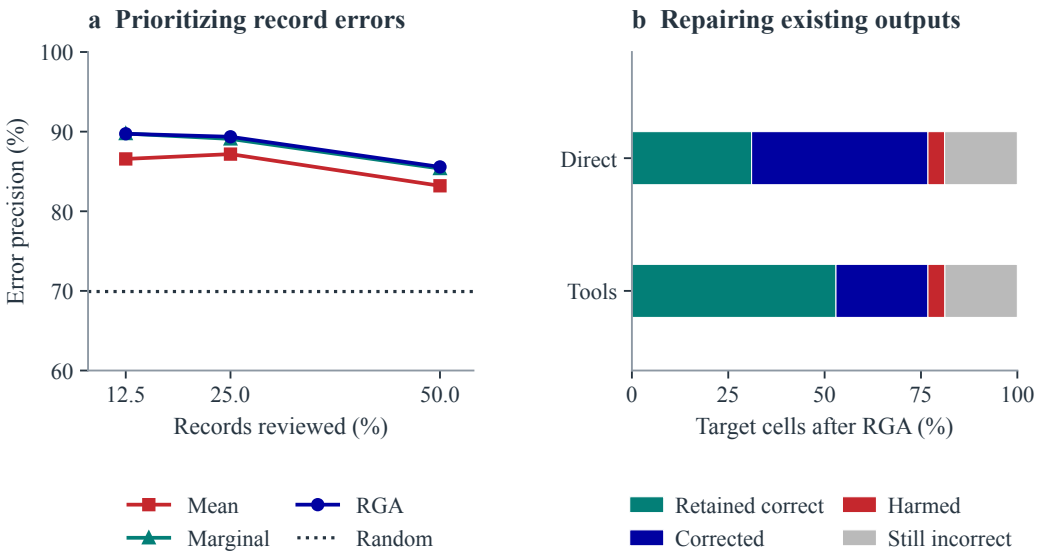


Figure S3: **Review and repair from the same compatibility information.** (a) Error precision across review budgets on common eligible controlled outputs. (b) Correctness transitions after reconstructing saved direct and tool outputs on controlled candidates. Review uses assigned-record scores; reconstruction solves the complete assignment. Both use LimiX-2 compatibility, with the distinct supports described in the text.

Marginal density reaches 77.90% cell accuracy; TabPFN-3.5 reaches 66.00% with mean residual and 75.86% with marginal density. These scores refer to statistical association, not the combined 512-episode set.

Table S12 reports the three LimiX-2 scoring configurations by document evidence. Their 79.59% RGA identity-stratum accuracy uses statistical compatibility with document-key constraints disabled. The 100% inventory ceiling is a separate gold-informed diagnostic (Table S3). Extension membership and scorer configurations were fixed before inspecting extension results.

The RecordBench evaluations use related reference rows, fixed candidate inventories, two numeric targets and a closed one-to-one assignment. The Text2DB evaluation uses partial assignment and observed numeric target coordinates under the same compatibility formulation. Incorrect or absent extracted values remain part of the output inventory and limit what reassignment can recover. Explicit document keys provide direct identity where available. Open-ended extraction, candidate creation and arbitrary relational schemas are different operations from the measured record-formation task.

The Text2DB comparison in Table 4 evaluates the current LimiX-2 RGA on a separate reference-eligible cohort; its protocol and complete denominators appear in Appendix S9.

S7.1 Complete point estimates and component variation

Table S13: **Primary cross-setting point estimates.** Target-cell and complete-record accuracies are percentages. All rows share references and candidates within a setting. The final three rows report the LimiX-2 mean and marginal ablations and complete RGA. Llama averages three seeds; Qwen uses one. Dashes denote unmeasured combinations. All planned episodes, including failures, remain in the reported cohorts.

Record-formation method	Controlled cell	Controlled record	Extracted cell	Extracted record
DocETL-Join	–	–	25.18	13.87
LOTUS-Join	–	–	3.09	1.57
LOTUS-Global	–	–	28.95	17.95
Qwen3.5-4B: direct	35.37	29.20	35.28	21.98
Qwen3.5-4B: generated matrix	21.57	15.10	23.13	11.63
Qwen3.5-4B: executed tools	57.35	52.28	51.48	37.03
Llama-3.1-8B: direct	24.51	18.09	–	–
Llama-3.1-8B: executed tools	57.29	51.80	–	–
Regression-CatBoost	58.22	52.85	51.49	37.05
Pair-CatBoost	64.53	59.37	57.13	42.05
ExtraTrees	59.97	54.88	52.61	38.56
TabICLv2: mean residual	64.44	59.44	56.27	41.78
TabPFN-3: mean residual	66.16	61.31	57.56	42.47
TabPFN-3.5: mean residual	68.32	63.74	60.12	45.01
TabPFN-3.5: marginal density	75.39	71.31	68.37	51.57
LimiX-2: mean residual	69.59	65.20	60.87	45.85
LimiX-2: marginal density	76.01	72.05	68.15	51.73
RGA with LimiX-2	**76.81**	**72.85**	**68.94**	**52.37**

S8 Larger Language Model

Qwen3.5-27B uses the same fixed 1,024-episode cohort, with 768 statistical-association episodes and 256 explicit-identity controls, and all four language pathways. Three complete executions are averaged within each episode before source-component aggregation. This comparison shares the frozen LimiX-2 reconstruction with the smaller-model comparisons.

Table S14: **Qwen3.5-27B capability and paired RGA improvement.** Accuracies are percentages; gains refer to controlled target-cell accuracy and include paired 95% component-bootstrap intervals. The explicit-identity condition is reported separately.

Pathway	Controlled cell	Controlled record	Identity cell	Identity record	RGA gain [95% interval]
Direct	44.68	38.87	96.88	96.88	+32.13 [27.97, 36.46]
Generated matrix	31.36	24.92	52.27	48.29	+45.45 [41.07, 49.99]
Executed tools	58.58	53.28	57.00	51.20	+18.23 [15.19, 21.27]
Self-refinement	44.13	39.02	96.22	96.22	+32.68 [28.49, 37.16]

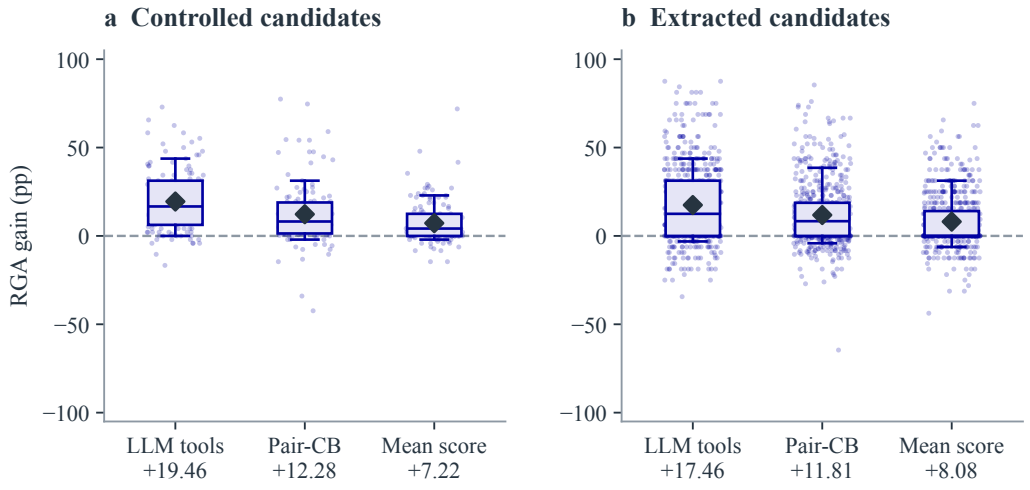


Figure S4: **Component-level RGA gains.** Every dot is one source component, after averaging its episodes: 128 controlled and 605 extracted components. Boxes show the 25th, 50th and 75th percentiles; whiskers show the 10th and 90th percentiles. Black diamonds and numbers below each group give the mean gain. The plot shows the full component-gain distribution; jitter changes only the horizontal position. The reference model, observed candidate bank and evaluated support are fixed within each paired comparison.

S9 Database Integration under Partial Assignment

S9.1 Task support and extraction budgets

Eligible Text2DB infilling/row-population tasks, selected from instructions and initial databases, specify one unambiguous target table with a stable, unique, non-null identity key. Infilling requires an entirely numeric missing-column block excluding the key. Row population targets all existing non-key numeric columns; other columns remain anchors. Each target requires at least 16 finite initial observations and no nonnumeric values. Column addition, ambiguous tables and composite keys are excluded.

The development partition contains 9 tasks from 6 source components; the evaluation partition contains 23 tasks from 22 different components. All budgets use these same partitions, which are separate from RecordBench sources. Reference records come from the initial database, with query-entity overlap removed before scoring. The saved Qwen3.5-4B extraction seeds are 9152026, 9152027 and 9152028. Budget one evaluates each run separately. Budget two evaluates all three run-pair unions, $(1, 2)$, $(1, 3)$ and $(2, 3)$. Budget three evaluates the union of all three runs. Pool construction uses the observed inventories and schema, without answer access.

Each method is calibrated separately for each budget on the matching development inventories. Budget-two calibration averages all three pairs to select one threshold. The selected threshold is fixed for that budget's evaluation. All three calibrations yield the same method-specific thresholds. Table S15 records the inventory units and extraction support.

Table S15: **Input inventories by extraction budget.** Budget-one counts are task–extraction banks; budget-two counts are task–pair unions; budget-three counts are task-level three-run unions. Scorer-seed copies do not add inventories. Empty-side and missing-target rows count completed inventories; numeric targets must be finite under the scorer's parser. All inventories remain in the reported outcomes.

Quantity	Dev. 1	Eval. 1	Dev. 2	Eval. 2	Dev. 3	Eval. 3
Tasks	9	23	9	23	9	23
Source components	6	22	6	22	6	22
Distinct inventory units	27	69	27	69	9	23
Completed inventories	25	64	26	65	9	22
Extraction-failed inventories	2	5	1	4	0	1
Completed inventories with an empty side	9	15	9	15	3	5
Completed inventories with nonempty sides but no finite numeric target	6	15	6	15	2	5
At least two anchors and two candidates	4	9	5	9	2	3

S9.2 Anchor alignment and candidate pooling

Anchors are grouped by normalized values in the schema-defined identity columns. If an identity is absent, the key falls back to the complete anchor-value tuple. The most frequent observed representation is retained in each group, with lexicographic tie breaking over serialized cells, identity and evidence. Source-provided anchors retain their frozen identity and provenance.

Candidates are grouped by their entire normalized target-value tuple, including missing positions. If tuple v occurs $m_r(v)$ times in run r, the union retains $\max_r m_r(v)$ separately indexed occurrences. Numeric normalization uses finite decimal values where parsing succeeds; textual normalization applies Unicode NFKC, case folding and whitespace normalization. Missing values are represented explicitly. Candidate identity and evidence select a deterministic representative but are not part of the grouping key. Equal-valued records retain the maximum within-run multiplicity; the rule does not resolve entity-specific duplicates across runs. Distinct value hypotheses and complementary partial tuples remain separate candidates, without combining their values to fill missing coordinates.

Each anchor receives at most one candidate occurrence, each occurrence is used at most once, and an anchor may remain unmatched at the calibrated cost. Source constraints are preserved; this cohort has no explicit fixed or forbidden edges. Materialization copies observed values from the selected candidate and does not synthesize missing values.

S9.3 Scorer configurations

LimiX Mean uses the saved epoch-20 point-prediction checkpoint. Joint RGA and Marginal share the architecture-2 epoch-1110 density checkpoint, with inference seed 9182026 and softmax temperature 0.9. The density computation uses a uniform bin-mass mixture of 10^{-6} and float64 evaluation. TabPFN uses its saved v3 regression checkpoint with eight estimators; Pair-CatBoost uses depth 6 and 400 iterations. Exact configuration files accompany the reproduction archive.

Candidates are grouped by observed numeric coordinates; references complete on those coordinates supply target means and scales. Mean-residual, marginal-density and bidirectional joint-density costs use these coordinates without imputation. Candidates without numeric observations have no statistical edge. Empty inventories yield empty assignments. The forward/reverse Joint mixture weights are fixed at 0.5/0.5.

S9.4 Threshold search and paired one-standard-error rule

We use a paired source-component one-standard-error threshold-selection rule (Hastie et al., 2009). The search sets reflect each method's cost scale (Table S16). For a given method and extraction

budget, let $F_c(t)$ be development changed-cell F1 at threshold t, averaged within each task over its run-subset and scorer-seed slots, then across tasks in source component c. With $C = 6$ development components, define

$$t^* = \max \arg\max_{t \in \mathcal{G}} \frac{1}{C} \sum_{c=1}^{C} F_c(t), \qquad L_c(t) = F_c(t^*) - F_c(t).$$

Exact ties for the development optimum favor the larger threshold. For each threshold, compute

$$\overline{L}(t) = \frac{1}{C} \sum_{c=1}^{C} L_c(t), \qquad \mathrm{SE}_{\mathrm{paired}}(t) = \sqrt{\frac{1}{C(C-1)} \sum_{c=1}^{C} \big(L_c(t) - \overline{L}(t)\big)^2}.$$

The selected threshold is

$$\widehat{t} = \max \left\{ t \in \mathcal{G} : \overline{L}(t) \leq \mathrm{SE}_{\mathrm{paired}}(t) + 10^{-12} \right\}.$$

The tolerance is numerical; the standard error uses paired component-level losses. A larger unmatched cost favors matching. This is a selection heuristic, not a significance test. The rule uses changed-cell F1 throughout.

Table S16: **Threshold search by cost family.** Each method is calibrated independently at each extraction budget; the selected cost is the same at budgets one, two and three.

Method	Threshold grid	Selected
Joint RGA; Marginal	$\{-32, -16, -8, -4, -2, -1, 0, 1, 2, 4, 8, 16, 32\}$	4
LimiX Mean; TabPFN	$\{0.25, 1, 4, 16, 64\}$	64
Pair-CatBoost	$\{-4, -2, 0, 2, 4\}$	4

S9.5 Database outcomes, aggregation and failures

From the initial database, matches copy observed target cells; unmatched anchors preserve existing values. Infilling updates identities; row population can append extracted new identities. SQLite affinities precede comparison of changed non-null facts keyed by table, identity and column, following Text2DB semantics for this stable-schema, single-key subset.

Task-level scores average all prescribed run-subset and scorer-seed slots. Budget one has three slots per task. Budget two has three extraction pairs, each represented by three scorer-seed slots, giving nine slots per task. Budget three has one pooled inventory represented by three slots. Point scorers use saved seed-specific costs. For pooled density scoring, one deterministic result per pool is repeated in three slots for aggregation compatibility. Task averages are averaged within components, and the 22 components receive equal weight. Task F1 instead weights the 23 task averages equally; Association F1 uses the component-macro aggregation. The resulting 69, 207 and 69 records per method do not increase the number of independent sampling units.

All prescribed slots remain in the denominator. A failed extraction contributes its saved empty inventory; successful runs may supply candidates to a union containing a failed run. The evaluation has five extraction-failed inventories at budget one, four at budget two and one at budget three; all corresponding method outputs have zero changed-cell F1. Completed inventories with empty sides or no finite numeric targets are also retained.