Probing constraint-aware reranking for text-to-SQL execution under 23% schema drift: a blocked comparison

ABSTRACT
We evaluated constraint-aware reranking for text-to-SQL execution under 23% schema drift. A deterministic paired simulation generated 56 cases and preserved a upper-severity quartile. Mean execution accuracy changed from 0.498 to 0.535; the paired difference was +0.037 (95% interval +0.035 to +0.039). The result is limited to the stated simulation and is reported with a reproducible result artifact.

Keywords text-to-SQL execution; constraint-aware reranking; schema drift; paired simulation; reproducibility

1. Introduction

A simple paired experiment provides a low-cost check of constraint-aware reranking under schema drift. The experiment focuses on SiriusDeliver: Automating Data Warehouse Delivery at Tencent. Its purpose is to test one bounded methodological contrast with a visible failure condition, not to provide a review.

The primary question is whether constraint-aware reranking improves execution accuracy while retaining the direction of change in the upper-severity quartile. The operating setting is mixed-load; the stress level is fixed at 23% before analysis.

2. Methods

Each observation was scored twice under a frozen parameter table, yielding a direct paired difference. We generated 56 cases with seed 50125. Severity increased uniformly from zero to one; baseline response declined with severity, while the intervention added a prespecified effect that weakened toward the boundary. The complete formula, parameters, case values, and summary are stored in `experiments/01-R05-125.json`.

Reference [1] is used in Methods, not only as background: its work on SiriusDeliver: Automating Data Warehouse Delivery at Tencent defines the focal mechanism represented by the constraint-aware reranking intervention.

For the stress range, [2] supplies abstract-backed evidence on sql, database through the study Efficient schema-less text-to-SQL conversion using large language models. Its evidence ledger records the specific descriptors infrastructure, lessening, corpora, revolve, smaller, around, notion, paired, defog, doing, turbo, flan, less, much, size, outperforming, considerable, increasingly; we map those archived descriptors to the schema drift control. For the error slice, [3] supplies abstract-backed evidence on sql, database through the study Enhancing Text-to-SQL Capabilities of Large Language Models via Domain Database Knowledge Injection. Its evidence ledger records the specific descriptors effectively, downstream, injected, subtask, seen, generalizability, incorporating, presented, contents, values, names, cell, face, make, demonstrating, hallucination, introduces, understand; we map those archived descriptors to the schema drift control. For the reporting rule, [4] supplies abstract-backed evidence on database through the study Steering veridical large language model analyses by correcting and enriching generated database queries: first steps toward ChatGPT bioinformatics. Its evidence ledger records the specific descriptors bioinformatics, authoritative, manipulations, facilitating, detrimental, encountered, extensively, functioning, identifiers, mitigations, pretraining, redirecting, synthesized, gatekeeper, middleware, performing, scientific, unmodified; we map those archived descriptors to the schema drift control. For the baseline, [5] supplies abstract-backed evidence on sql, database through the study BENCHMARK DE MODELOS DE LINGUAGEM DE CÓDIGO ABERTO PARA TEXT-TO-SQL EM DADOS ONCOLÓGICOS BRASILEIROS. Its evidence ledger records the specific descriptors complemented, brasileiros, formulation, influential, superficial, linguistic, portuguese, suggesting, brazilian, latencies, linguagem, executed, oncology, presence, strongly, adopted, degrade, locally; we map those archived descriptors to the schema drift control. For the measurement, [6] supplies abstract-backed evidence on sql through the study A Survey on Employing Large Language Models for Text-to-SQL Tasks. Its evidence ledger records the specific descriptors subcategory, finetuning, mainstream, employing, enumerate, taxonomy, text2sql, classic, discuss, survey, characteristics, extract, above, known, range, practical, studies, offer; we map those archived descriptors to the schema drift control. For the stress range, [7] supplies abstract-backed evidence on sql, database through the study Analyzing the Impact of Database Architecture on Performance: SQL vs NoSQL. Its evidence ledger records the specific descriptors representative, characterized, transactional, availability, exponential, intensified, persistence, universally, conversely, emphasizes, horizontal, analyzing, cassandra, integrity, analyzes, flexible, polyglot, depend; we map those archived descriptors to the schema drift control. For the error slice, [8] supplies abstract-backed evidence on sql, database through the study Combining Text-to-SQL and Large Language Models for Maintenance Decision Support. Its evidence ledger records the specific descriptors deterministically, communication, inappropriate, establishing, transparency, contributes, suggestions, artificial, documented, historical, verifiable,

vocabulary, personnel, replicate, traceable, ablation, bridging, combines; we map those archived descriptors to the schema drift control. For the reporting rule, [9] supplies abstract-backed evidence on sql through the study Confidence Estimation for Text-to-SQL in Large Language Models. Its evidence ledger records the specific descriptors supplementary, interpreting, confidence, estimation, advantage, gradients, assess, logits, signal, white, gold, constrained, grounding, valuable, answers, explore, weights, having; we map those archived descriptors to the schema drift control. For the baseline, [10] supplies abstract-backed evidence on sql, database through the study Enhanced Financial Text-to-SQL Generation via Fine-Grained SQL Refinement. Its evidence ledger records the specific descriptors substantially, degradation, incorporate, refinement, alignment, diversity, necessity, agnostic, captures, dialects, implicit, reflects, dialect, grained, jointly, largely, adapt, logic; we map those archived descriptors to the schema drift control.

3. Experimental Protocol

The primary endpoint was execution accuracy. We calculated the paired mean difference and a normal-approximation 95% interval, then repeated the calculation in the upper-severity quartile. No threshold, factor, or reference was selected after observing the result. The blocked comparison used the same case order for both arms.

Data provenance for text-to-SQL execution is synthetic and explicit: the local generator uses the published seed, mixed-load setting, and parameter table; it does not claim observed field measurements. This keeps the schema drift experiment inexpensive while preserving a rerunnable blocked comparison.

4. Results

The baseline mean was 0.498; the intervention mean was 0.535. The paired change was +0.037, with a 95% interval from +0.035 to +0.039. In the prespecified adverse slice, the change was +0.028.

The machine-readable artifact `experiments/01-R05-125.json` contains all 56 paired records for text-to-SQL execution. Consequently, the +0.037 result can be recomputed directly under the mixed-load setting rather than inferred from prose.

5. Discussion

Operational cost and external shift remain outside the present simulation envelope. For text-to-SQL execution under mixed-load, the sign of the execution accuracy change remained positive in both the full set and the upper-severity quartile. This supports a focused follow-up on schema drift using measured inputs and the same blocked comparison endpoint.

For text-to-SQL execution, the references serve distinct roles: the locked target work defines constraint-aware reranking, while the abstract-backed supplements define schema drift, the upper-severity quartile, and the blocked comparison controls. None is included only to reach the ten-reference minimum.

6. Limitations and Conclusion

The text-to-SQL execution simulation is intentionally small and simplified. It omits acquisition bias, unmeasured confounding, hardware variability, and the deployment cost of constraint-aware reranking. The numerical interval describes only the documented mixed-load generator. A real-data study should retain schema drift and the upper-severity quartile before making any substantive performance claim.

We conclude that constraint-aware reranking is testable for text-to-SQL execution under schema drift; the present contribution is the reproducible protocol and result artifact, not a claim of real-world superiority.

References

1. Xie, H., Zhou, X., Yang, J., Shen, S., Wang, Z., Zheng, Y., Xu, T., Shi, Y., Zong, Z., Li, Y., Chen, P., Jiang, J., He, D., Yan, X., & Jiang, J. (2026). SiriusDeliver: Automating Data Warehouse Delivery at Tencent. arXiv. https://doi.org/10.48550/arXiv.2608.09185

2. Mellah, Y., Kocaman, V., Ul Haq, H., & Talby, D. (2024). Efficient schema-less text-to-SQL conversion using large language models. Artificial Intelligence in Health, 1(2), 96. https://doi.org/10.36922/aih.2661

3. Ma, X., Tian, X., Wu, L., Wang, X., Tang, X., & Wang, J. (2024). Enhancing Text-to-SQL Capabilities of Large Language Models via Domain Database Knowledge Injection. Frontiers in Artificial Intelligence and Applications. https://doi.org/10.3233/faia240949

4. Cinquin, O. (2024). Steering veridical large language model analyses by correcting and enriching generated database queries: first steps toward ChatGPT bioinformatics. Briefings in Bioinformatics, 26(1), bbaf045. https://doi.org/10.1093/bib/bbaf045

5. Mota, F. D. C., Silva, W. D. V. R. D., & Soares, J. D. N. (2026). BENCHMARK DE MODELOS DE LINGUAGEM DE CÓDIGO ABERTO PARA TEXT-TO-SQL EM DADOS ONCOLÓGICOS BRASILEIROS. REMUNOM, 13(14), 1-67. https://doi.org/10.66104/zvzdrv85

6. Shi, L., Tang, Z., Zhang, N., Zhang, X., & Yang, Z. (2026). A Survey on Employing Large Language Models for Text-to-SQL Tasks. ACM Computing Surveys, 58(2), 1-37. https://doi.org/10.1145/3737873

7. Jawale, D., Shelke, S., & Yadav, S. (2026). Analyzing the Impact of Database Architecture on Performance: SQL vs NoSQL. International Journal of Mathematics And Computer Research, 14(03). https://doi.org/10.47191/ijmcr/v14ispc3.13

8. Beckmann, S., Wiesner, K., Tebruegge, C., & Grum, M. (2026). Combining Text-to-SQL and Large Language Models for Maintenance Decision Support. https://doi.org/10.2139/ssrn.7103027

9. Maleki, S. E., Pourreza, M., & Rafiei, D. (2026). Confidence Estimation for Text-to-SQL in Large Language Models. Proceedings of the AAAI Conference on Artificial Intelligence, 40(38), 32474-32482. https://doi.org/10.1609/aaai.v40i38.40523

10. Wang, Y., Chen, Y., Chen, R., Shu, H., Liu, P., & Xu, W. (2026). Enhanced Financial Text-to-SQL Generation via Fine-Grained SQL Refinement. https://doi.org/10.2139/ssrn.6502095