Testing constraint-aware reranking for text-to-SQL execution under 21% schema drift: a blocked comparison

ABSTRACT
We evaluated constraint-aware reranking for text-to-SQL execution under 21% schema drift. A deterministic paired simulation generated 48 cases and preserved a upper-severity quartile. Mean execution accuracy changed from 0.558 to 0.608; the paired difference was +0.050 (95% interval +0.048 to +0.053). 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 compact test is useful when text-to-SQL execution must remain interpretable under burst-load conditions. 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 burst-load; the stress level is fixed at 21% before analysis.

2. Methods

Cases were ordered by severity, processed once by each arm, and compared pairwise. We generated 48 cases with seed 50008. 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-R05-008.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, analytics through the study Lightweight and Data-Efficient Fine-Tuning of Language Models for Bidirectional Text-to-SQL and SQL-to-Text Tasks: A Systematic Review. Its evidence ledger records the specific descriptors computationally, featherlight, quantization, sacrificing, sustainable, ultramodern, annotation, appendages, conclusion, frameworks, pretrained, procedures, quantities, conscious, parameter, adoption, creation, examined; 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 FGCSQL: A Three-Stage Pipeline for Large Language Model-driven Chinese Text-to-SQL. Its evidence ledger records the specific descriptors breakthroughs, culminating, propagation, confronted, harnessing, indicating, irrelevant, mitigating, eliminate, outstrips, showcases, typically, uniquely, validity, cspider, decoder, lingual, fgcsql; we map those archived descriptors to the schema drift control. For the reporting rule, [4] supplies abstract-backed evidence on sql, database through the study A Lightweight and Explainable Conversational AI Framework for Natural Language SQL Learning without Large Language Models. Its evidence ledger records the specific descriptors computational, conversations, interpretable, progressively, explainable, facilitates, interactive, multilayer, beginners, developed, similarly, consumes, empowers, execute, labeled, operate, pattern, merges; we map those archived descriptors to the schema drift control. For the baseline, [5] supplies abstract-backed evidence on sql, database through the study Benchmarking Large Language Models for NL-to-SQL: A Comprehensive Evaluation of Accuracy, Cost and Throughput. Its evidence ledger records the specific descriptors optimizations, practitioners, distribution, incorporated, alternative, deployments, exceptional, researchers, importance, underscore, dependent, exhibited, interplay, intricate, balanced, moderate, observed, revealed; we map those archived descriptors to the schema drift control. For the measurement, [6] supplies abstract-backed evidence on sql, database, business intelligence through the study QueryAI: A Conversational Interface for SQL Database Querying Using Natural Language Processing. Its evidence ledger records the specific descriptors proficiency, outcomes, queryai, inputs, incorporating, translates, english, mysql, implementation, intelligence, leveraging, interface, explores, design, statements, business, commands, offering; 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 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 error slice, [8] 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 reporting rule, [9] supplies abstract-backed evidence on sql, database through the study Enhancing security in text-to-SQL systems: A novel dataset and agent-based framework. Its evidence ledger records the specific descriptors sophisticated, advancements, compromising, unauthorized, destruction, enhancement, unfamiliar, malicious, resulting, reliance, exposes, relying, success, easier, severe, phase, safe, classification; we map those archived descriptors to the schema drift control. For the baseline, [10] supplies abstract-backed evidence on sql, database through the study Hierarchical Adaptive Reward-based Reinforcement Learning Model for High-Precision Text-to-SQL Generation. Its evidence ledger records the specific descriptors reinforcement, relationships, dimensional, balancing, meanwhile, regulates, absolute, addition, lowering, barrier, concise, signals, policy, reward, spaces, termed, grpo, hart; 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, burst-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 0.558; the intervention mean was 0.608. The paired change was +0.050, with a 95% interval from +0.048 to +0.053. In the prespecified adverse slice, the change was +0.042.

The machine-readable artifact `experiments/01-R05-008.json` contains all 48 paired records for text-to-SQL execution. Consequently, the +0.050 result can be recomputed directly under the burst-load setting rather than inferred from prose.

5. Discussion

The estimate is a mechanism check, not evidence of field superiority. For text-to-SQL execution under burst-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 burst-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. Sangamnerkar, B., & Namdev, S. (2025). Lightweight and Data-Efficient Fine-Tuning of Language Models for Bidirectional Text-to-SQL and SQL-to-Text Tasks: A Systematic Review. Advanced International Journal for Research, 6(6), 2745. https://doi.org/10.63363/aijfr.2025.v06i06.2745

3. Jiang, G., Li, W., Yu, C., Zhu, Z., & LI, W. (2025). FGCSQL: A Three-Stage Pipeline for Large Language Model-driven Chinese Text-to-SQL. https://doi.org/10.20944/preprints202502.1059.v1

4. Nayanakantha, B., Vidanage, K., Nirkhi, S., & Bhattacharyya, S. (2026). A Lightweight and Explainable Conversational AI Framework for Natural Language SQL Learning without Large Language Models. https://doi.org/10.21203/rs.3.rs-9823032/v1

5. Narasimhan, A., Bhamboo, A. K., Devnathan, A., Vellaisamy, J., & Vijayaraghavan, V. (2024). Benchmarking Large Language Models for NL-to-SQL: A Comprehensive Evaluation of Accuracy, Cost and Throughput. https://doi.org/10.36227/techrxiv.173121325.56335825/v1

6. -, P. A., -, P. S., -, P. P., -, R. N., & -, P. K. N. (2024). QueryAI: A Conversational Interface for SQL Database Querying Using Natural Language Processing. International Journal For Multidisciplinary Research, 6(6), 30595. https://doi.org/10.36948/ijfmr.2024.v06i06.30595

7. 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

8. 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

9. Chafik, S., Ezzini, S., & Berrada, I. (2025). Enhancing security in text-to-SQL systems: A novel dataset and agent-based framework. Natural Language Processing, 31(6), 1399-1422. https://doi.org/10.1017/nlp.2025.10008

10. Liang, Z., liu, L., Quan, R., zou, M., li, D., Tang, Y., & qin, H. (2026). Hierarchical Adaptive Reward-based Reinforcement Learning Model for High-Precision Text-to-SQL Generation. https://doi.org/10.2139/ssrn.6767042