Testing constraint-aware reranking for text-to-SQL execution under 9% schema drift: a factor ablation

ABSTRACT
We evaluated constraint-aware reranking for text-to-SQL execution under 9% schema drift. A deterministic paired simulation generated 48 cases and preserved a low-signal stratum. Mean execution accuracy changed from 0.558 to 0.596; the paired difference was +0.038 (95% interval +0.036 to +0.041). 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 low-signal stratum. The operating setting is burst-load; the stress level is fixed at 9% before analysis.

2. Methods

Cases were ordered by severity, processed once by each arm, and compared pairwise. We generated 48 cases with seed 50080. 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-R05-080.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 through the study CogSQL: A Cognitive Framework for Enhancing Large Language Models in Text-to-SQL Translation. Its evidence ledger records the specific descriptors transitional, composition, preparation, replicating, correction, prediction, cognitive, featuring, mirroring, processes, applying, concepts, consists, drafting, holistic, refining, aspects, believe; we map those archived descriptors to the schema drift control. For the error slice, [3] supplies abstract-backed evidence on database through the study ChIP-GPT: a managed large language model for robust data extraction from biomedical database records. Its evidence ledger records the specific descriptors immunoprecipitation, comprehensively, identification, fundamentally, characterize, emphasizing, iteratively, customized, predefined, seamlessly, adaptable, chromatin, hindering, amassing, centered, medicine, reliably, archive; 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 MedT5SQL: a transformers-based large language model for text-to-SQL conversion in the healthcare domain. Its evidence ledger records the specific descriptors difficulties, encountering, transformers, approximate, accessible, delivering, electronic, optimizers, prevalence, expertise, assesses, commonly, medt5sql, transfer, utilizes, medical, numbers, related; we map those archived descriptors to the schema drift control. For the baseline, [5] supplies abstract-backed evidence on sql, database through the study Large Language Model Method as a Translator Indonesian Into SQL Language. Its evidence ledger records the specific descriptors padangsidimpuan, administrative, automatically, intuitively, background, encouraged, supporting, translated, translator, lecturers, flexibly, intended, obstacle, becomes, certain, command, massive, quickly; we map those archived descriptors to the schema drift control. For the measurement, [6] supplies abstract-backed evidence on sql through the study Pengembangan Aplikasi Chatbot dengan Large Language Model untuk Text-to-SQL Generation. Its evidence ledger records the specific descriptors pengeksekusian, menerjemahkan, mengembangkan, menunjukkan, menyediakan, pemeriksaan, pemrograman, acceptable, pengecekan, pertanyaan, antarmuka, pemilihan, penulisan, pertanian, usability, aplikasi, kegunaan, kriteria; 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 AN ENHANCED NATURAL LANGUAGE PROCESSING MODEL FOR DATA EXTRACTION AND VISUALIZATION FROM SQL DATABASE STRUCTURES: A SYSTEMATIC REVIEW OF LITERATURE. Its evidence ledger records the specific descriptors visualization, fundamentals, underpinning, foundations, proceedings, synthesizes, conference, identifies, prototypes, resolution, trajectory, scholarly, journals, motivate, reviewed, draws, flash, maps; we map those archived descriptors to the schema drift control. For the error slice, [8] supplies abstract-backed evidence on sql through the study SQL/PGQ data model and graph schema. Its evidence ledger records the specific descriptors describing, documents, property, neo4j,

mapping, graph, three, schema; 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 GRASP-SQL: Graph Retrieval and Agentic Schema Pruning for Recall-First Text-to-SQL. Its evidence ledger records the specific descriptors discriminability, discriminatively, representational, preprocessing, statistically, concatenates, connectivity, conditioned, unnecessary, adaptively, ensembling, orthogonal, recruiting, ablations, embedding, expansion, generator, necessary; 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 low-signal stratum. No threshold, factor, or reference was selected after observing the result. The factor ablation 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 factor ablation.

4. Results

The baseline mean was 0.558; the intervention mean was 0.596. The paired change was +0.038, with a 95% interval from +0.036 to +0.041. In the prespecified adverse slice, the change was +0.030.

The machine-readable artifact `experiments/01-R05-080.json` contains all 48 paired records for text-to-SQL execution. Consequently, the +0.038 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 low-signal stratum. This supports a focused follow-up on schema drift using measured inputs and the same factor ablation 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 low-signal stratum, and the factor ablation 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 low-signal stratum 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. Yuan, H., Tang, X., Chen, K., Shou, L., Chen, G., & Li, H. (2025). CogSQL: A Cognitive Framework for Enhancing Large Language Models in Text-to-SQL Translation. Proceedings of the AAAI Conference on Artificial Intelligence, 39(24), 25778-25786. https://doi.org/10.1609/aaai.v39i24.34770

3. Cinquin, O. (2024). ChIP-GPT: a managed large language model for robust data extraction from biomedical database records. Briefings in Bioinformatics, 25(2), bbad535. https://doi.org/10.1093/bib/bbad535

4. Marshan, A., Almutairi, A. N., Ioannou, A., Bell, D., Monaghan, A., & Arzoky, M. (2024). MedT5SQL: a transformers-based large language model for text-to-SQL conversion in the healthcare domain. Frontiers in Big Data, 7, 1371680. https://doi.org/10.3389/fdata.2024.1371680

5. Putra, C., Arlis, S., & Nurcahyo, G. W. (2025). Large Language Model Method as a Translator Indonesian Into SQL Language. Jurnal KomtekInfo, 124-130. https://doi.org/10.35134/komtekinfo.v12i3.658

6. Nugraha, G. P., Suadaa, L. H., Wilantika, N., & Maghfiroh, L. R. (2024). Pengembangan Aplikasi Chatbot dengan Large Language Model untuk Text-to-SQL Generation. Seminar Nasional Official Statistics, 2024(1), 831-840. https://doi.org/10.34123/semnasoffstat.v2024i1.2252

7. C. Joel, U., & Dewole A. Temilola, J. (2025). AN ENHANCED NATURAL LANGUAGE PROCESSING MODEL FOR DATA EXTRACTION AND VISUALIZATION FROM SQL DATABASE STRUCTURES: A SYSTEMATIC REVIEW OF LITERATURE. Jana Nexus: Journal of Computer Science, 01(12), 26-31. https://doi.org/10.21474/jncs01/111

8. Furniss, P., & Green, A. (2023). SQL/PGQ data model and graph schema. https://doi.org/10.54285/ldbc.qzsk3559

9. Kim, H., Kim, W., & Kim, W. (2026). GRASP-SQL: Graph Retrieval and Agentic Schema Pruning for Recall-First Text-to-SQL. https://doi.org/10.2139/ssrn.7194451

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