

Probing constraint-aware reranking for text-to-SQL execution under 40% schema drift: a robust mean contrast

ABSTRACT

We evaluated constraint-aware reranking for text-to-SQL execution under 40% schema drift. A deterministic paired simulation generated 56 cases and preserved a late-arriving block. Mean execution accuracy changed from 0.546 to 0.601; the paired difference was +0.055 (95% interval +0.052 to +0.057). 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 late-arriving block. The operating setting is shifted-distribution; the stress level is fixed at 40% 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 50109. 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-R05-109.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 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 reporting rule, [4] supplies abstract-backed evidence on sql, database through the study Heuristic-Guided Text-to-SQL Translation with LLMs: Optimizing Natural Language Interfaces for Relational Databases. Its evidence ledger records the specific descriptors deteriorates, translations, corrections, engineered, optimizing, heuristic, modular, loops, opensource, rewriting, feedback, provided, mapping, mondial, plays, highlighting, promising, adaptive; we map those archived descriptors to the schema drift control. For the baseline, [5] supplies abstract-backed evidence on sql, database through the study Pengembangan Model Migrasi Database Relational ke NoSQL Memanfaatkan Metadata SQL. Its evidence ledger records the specific descriptors dikembangkan, memanfaatkan, transformasi, berdasarkan, penyimpanan, terstruktur, diperlukan, diterapkan, eksperimen, mengajukan, pendekatan, perbandingan, dilakukan, kecepatan, kelemahan, melakuakn, menyimpan, merupakan; we map those archived descriptors to the schema drift control. For the measurement, [6] 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 stress range, [7] supplies abstract-backed evidence on sql, database through the study Advancing Conversational Text-to-SQL: Context Strategies and Model Integration with Large Language Models. Its evidence ledger records the specific descriptors concatenation, interactively, competitive, individual, advancing, averaging, spotlight, boosting, extends, history, merging, observe, obtains, promise, avenue, codes, cosql, sparc; 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 Nabil: A Text-to-SQL Model Based on Brain-Inspired Computing Techniques and Large Language Modeling. Its evidence ledger records the specific descriptors discriminative,

spatiotemporal, normalization, abstraction, inevitable, champion, encoding, inspired, networks, verified, engines, entered, spiking, duckdb, fuses, nabil, intelligent, outperforms; 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 Querying Databases with Natural Language: The use of Large Language Models for Text-to-SQL tasks. Its evidence ledger records the specific descriptors descriptions, dissertation, openly, poorly, views, evaluates, although, confirms, enhances, involves, mondial, simpler, joins, known, experiment, friendly, struggle, given; we map those archived descriptors to the schema drift control. For the baseline, [10] 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.

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 late-arriving block. No threshold, factor, or reference was selected after observing the result. The robust mean contrast 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, shifted-distribution setting, and parameter table; it does not claim observed field measurements. This keeps the schema drift experiment inexpensive while preserving a rerunnable robust mean contrast.

4. Results

The baseline mean was 0.546; the intervention mean was 0.601. The paired change was +0.055, with a 95% interval from +0.052 to +0.057. In the prespecified adverse slice, the change was +0.044.

The machine-readable artifact `experiments/01-R05-109.json` contains all 56 paired records for text-to-SQL execution. Consequently, the +0.055 result can be recomputed directly under the shifted-distribution 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 shifted-distribution, the sign of the execution accuracy change remained positive in both the full set and the late-arriving block. This supports a focused follow-up on schema drift using measured inputs and the same robust mean contrast 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 late-arriving block, and the robust mean contrast 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 shifted-distribution generator. A real-data study should retain schema drift and the late-arriving block 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/aiih.2661>
3. 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>
4. Petrola, L., Brayner, A., & Franco, W. (2025). Heuristic-Guided Text-to-SQL Translation with LLMs: Optimizing Natural Language Interfaces for Relational Databases. *Anais do XL Simpósio Brasileiro de Banco de Dados (SBBd 2025)*, 126-139. <https://doi.org/10.5753/sbbd.2025.247037>
5. Utomo, M. N. Y. (2020). Pengembangan Model Migrasi Database Relational ke NoSQL Memanfaatkan Metadata SQL. *Jurnal Teknologi Elektroika*, 4(2), 1. <https://doi.org/10.31963/elektroika.v4i2.2212>
6. 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>
7. Ascoli, B. G., & Choi, J. D. (2025). Advancing Conversational Text-to-SQL: Context Strategies and Model Integration with Large Language Models. *Future Internet*, 17(11), 527. <https://doi.org/10.3390/fi17110527>
8. Zhou, F., Hu, S., Du, X., Li, N., Zhou, T., Zhao, Y., Shang, S., Ling, X., & Zhu, H. (2025). Nabil: A Text-to-SQL Model Based on Brain-Inspired Computing Techniques and Large Language Modeling. *Electronics*, 14(19), 3910. <https://doi.org/10.3390/electronics14193910>
9. Nascimento, E. R. S., & Casanova, M. A. (2024). Querying Databases with Natural Language: The use of Large Language Models for Text-to-SQL tasks. *Anais Estendidos do XXXIX Simpósio Brasileiro de Banco de Dados (SBBd Estendido 2024)*, 196-201. https://doi.org/10.5753/sbbd_estendido.2024.240552
10. 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>