Auditing constraint-aware reranking for text-to-SQL execution under 39% schema drift: a paired bootstrap

ABSTRACT

We evaluated constraint-aware reranking for text-to-SQL execution under 39% schema drift. A deterministic paired simulation generated 72 cases and preserved a boundary-condition stratum. Mean execution accuracy changed from 0.536 to 0.573; the paired difference was +0.037 (95% interval +0.034 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

The design begins from a narrow failure mode in text-to-SQL execution rather than a broad literature survey. The experiment focuses on SiriusBI: A Comprehensive LLM-Powered Solution for Data Analytics in Business Intelligence. 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 boundary-condition stratum. The operating setting is noisy-input; the stress level is fixed at 39% before analysis.

2. Methods

We used a deterministic severity sweep and retained identical noise draws for the primary contrast. We generated 72 cases with seed 50115. 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-R05-115.json`.

Reference [1] is used in Methods, not only as background: its work on SiriusBI: A Comprehensive LLM-Powered Solution for Data Analytics in Business Intelligence 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 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 baseline, [5] 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 measurement, [6] 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 stress range, [7] supplies abstract-backed evidence on sql, database through the study A comparative evaluation of large language models for detecting SQL injection vulnerabilities in web applications. Its evidence ledger records the specific descriptors vulnerabilities, confidential, intractable, potentially, circumvent, considered, manipulate, popularity, codellama, detecting, prominent, technique, emerging, payloads, produced, stronger, boolean, created; 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 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 reporting rule, [9] supplies abstract-backed evidence on sql, database through the study Natural Language to SQL at Scale: Integrating OpenAI with Oracle Autonomous Database via SELECT AI. Its evidence ledger records the specific descriptors attributable, mygithublink, reproducible, evaluations, independent, integrating, establish, reflected, overhead, targeted, feature, measure, medium, target, tiers, democratizing, demonstrated, intelligent; we map those archived descriptors to the schema drift control. For the baseline, [10] 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.

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 boundary-condition stratum. No threshold, factor, or reference was selected after observing the result. The paired bootstrap 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, noisy-input setting, and parameter table; it does not claim observed field measurements. This keeps the schema drift experiment inexpensive while preserving a rerunnable paired bootstrap.

4. Results

The baseline mean was 0.536; the intervention mean was 0.573. The paired change was +0.037, with a 95% interval from +0.034 to +0.039. In the prespecified adverse slice, the change was +0.027.

The machine-readable artifact `experiments/01-R05-115.json` contains all 72 paired records for text-to-SQL execution. Consequently, the +0.037 result can be recomputed directly under the noisy-input setting rather than inferred from prose.

5. Discussion

Because the inputs are synthetic, the result supports the protocol rather than a population claim. For text-to-SQL execution under noisy-input, the sign of the execution accuracy change remained positive in both the full set and the boundary-condition stratum. This supports a focused follow-up on schema drift using measured inputs and the same paired bootstrap 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 boundary-condition stratum, and the paired bootstrap 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 noisy-input generator. A real-data study should retain schema drift and the boundary-condition 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. Jiang, J., Xie, H., Shen, S., Shen, Y., Zhang, Z., Lei, M., Zheng, Y., Li, Y., Li, C., Huang, D., Wu, Y., Zhang, W., Cui, B., & Chen, P. (2025). SiriusBI: A Comprehensive LLM-Powered Solution for Data Analytics in Business Intelligence. Proceedings of the VLDB Endowment, 18(12), 4860-4873. https://doi.org/10.14778/3750601.3750610

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. 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

5. -, 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

6. 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

7. Hairan, B., & Şahman, M. A. (2026). A comparative evaluation of large language models for detecting SQL injection vulnerabilities in web applications. PeerJ Computer Science, 12, e4015. https://doi.org/10.7717/peerj-cs.4015

8. 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

9. Mishra, S. K. (2026). Natural Language to SQL at Scale: Integrating OpenAI with Oracle Autonomous Database via SELECT AI. https://doi.org/10.36227/techrxiv.177281023.37874227/v1

10. 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