Testing constraint-aware reranking for text-to-SQL execution under 26% schema drift: a blocked comparison

ABSTRACT
We evaluated constraint-aware reranking for text-to-SQL execution under 26% schema drift. A deterministic paired simulation generated 48 cases and preserved a median slice. Mean execution accuracy changed from 0.483 to 0.537; the paired difference was +0.054 (95% interval +0.052 to +0.056). 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 sparse-observation conditions. The experiment focuses on ZhuJiu: A Multi-dimensional, Multi-faceted Chinese Benchmark for Large Language Models. 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 median slice. The operating setting is sparse-observation; the stress level is fixed at 26% before analysis.

2. Methods

Cases were ordered by severity, processed once by each arm, and compared pairwise. We generated 48 cases with seed 50064. 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-R05-064.json`.

Reference [1] is used in Methods, not only as background: its work on ZhuJiu: A Multi-dimensional, Multi-faceted Chinese Benchmark for Large Language Models 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 CRED-SQL: Enhancing Real-World Large Scale Database Text-to-SQL Parsing Through Cluster Retrieval and Execution Description. Its evidence ledger records the specific descriptors reformulation, alleviating, description, exacerbated, spiderunion, decomposes, ultimately, birdunion, deviation, mismatch, performs, pinpoint, cluster, smduan, stages, drift, cred, sota; we map those archived descriptors to the schema drift control. For the error slice, [3] supplies abstract-backed evidence on sql through the study Text-to-SQL Empowered by Large Language Models: A Benchmark Evaluation. Its evidence ledger records the specific descriptors investigations, disadvantages, additionally, explorations, organization, systematical, leaderboard, supervised, designing, elaborate, empowered, refreshes, economic, inhibits, inspires, compare, example, towards; 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 ETM: Modern Insights into Perspective on Text-to-SQL Evaluation in the Age of Large Language Models. Its evidence ledger records the specific descriptors stylistically, counterparts, misrepresent, overestimate, shortcomings, perspective, problematic, community, overlooks, elements, ignoring, inherent, negative, release, anyone, script, rates, rigid; we map those archived descriptors to the schema drift control. For the baseline, [5] 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 measurement, [6] 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 stress range, [7] supplies abstract-backed evidence on sql, database through the study A COMPARATIVE STUDY OF LLM - POWERED DATABASE INTERFACES VERSUS TRADITIONAL SQL SYSTEMS FOR INVENTORY MANAGEMENT. Its evidence ledger records the specific descriptors deterministic, unprecedented, prioritizing, emonstrate, guaranteed, llmpowered, sqlalchemy, identical, inventory, penalties, averaged, backends, degraded, parallel, mission, slower, versus, incur; 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 ETL pipelines and

SQL database management. Its evidence ledger records the specific descriptors indispensable, authenticate, consolidated, contemporary, continuously, centralized, elucidating, responsible, configured, immediate, processed, frontend, visitors, display, visitor, manner, serves, page; we map those archived descriptors to the schema drift control. For the reporting rule, [9] supplies abstract-backed evidence on sql, database, business intelligence through the study Natural Language to SQL (NL2SQL): A Comprehensive Study of Text-to-SQL Systems, Conversational AI for Databases, Enterprise Architectures, Challenges, and Future Directions. Its evidence ledger records the specific descriptors differentiator, pharmaceutical, orchestration, organizations, hallucinated, exemplified, illustrates, interrogate, responsibly, accumulate, components, embeddings, executives, glossaries, multimodal, prescriber, reflection, validators; we map those archived descriptors to the schema drift control. For the baseline, [10] 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.

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 median slice. 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, sparse-observation 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 0.483; the intervention mean was 0.537. The paired change was +0.054, with a 95% interval from +0.052 to +0.056. In the prespecified adverse slice, the change was +0.046.

The machine-readable artifact `experiments/01-R05-064.json` contains all 48 paired records for text-to-SQL execution. Consequently, the +0.054 result can be recomputed directly under the sparse-observation 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 sparse-observation, the sign of the execution accuracy change remained positive in both the full set and the median slice. 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 median slice, 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 sparse-observation generator. A real-data study should retain schema drift and the median slice 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. Zhang, B., Xie, H., Du, P., Chen, J., Cao, P., Chen, Y., Liu, S., Liu, K., & Zhao, J. (2023). ZhuJiu: A Multi-dimensional, Multi-faceted Chinese Benchmark for Large Language Models. Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing: System Demonstrations, 479-494. https://doi.org/10.18653/v1/2023.emnlp-demo.44

2. Duan, S., Wang, Z., Liu, C., Zhu, Z., Zhang, Y., Han, P., Yan, L., & Peng, Z. (2025). CRED-SQL: Enhancing Real-World Large Scale Database Text-to-SQL Parsing Through Cluster Retrieval and Execution Description. Frontiers in Artificial Intelligence and Applications. https://doi.org/10.3233/faia251337

3. Gao, D., Wang, H., Li, Y., Sun, X., Qian, Y., Ding, B., & Zhou, J. (2024). Text-to-SQL Empowered by Large Language Models: A Benchmark Evaluation. Proceedings of the VLDB Endowment, 17(5), 1132-1145. https://doi.org/10.14778/3641204.3641221

4. Ascoli, B. G., Kandikonda, Y. S. R., & Choi, J. D. (2025). ETM: Modern Insights into Perspective on Text-to-SQL Evaluation in the Age of Large Language Models. Future Internet, 17(8), 325. https://doi.org/10.3390/fi17080325

5. 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

6. 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

7. Guo, M., & Sun, Y. (2026). A COMPARATIVE STUDY OF LLM - POWERED DATABASE INTERFACES VERSUS TRADITIONAL SQL SYSTEMS FOR INVENTORY MANAGEMENT. NLP & Text Mining, 11-21. https://doi.org/10.5121/csit.2026.160602

8. Muppala, M. (2025). ETL pipelines and SQL database management. SQL Database Mastery: Relational Architectures, Optimization Techniques,and Cloud-Based Applications, 84-101. https://doi.org/10.70593/978-93-7185-191-6_5

9. Shamal Chavan and Prof. Sandeep Vishwakarma (2026). Natural Language to SQL (NL2SQL): A Comprehensive Study of Text-to-SQL Systems, Conversational AI for Databases, Enterprise Architectures, Challenges, and Future Directions. International Journal of Advanced Research in Science Communication and Technology, 380. https://doi.org/10.48175/ijarsct-37344

10. 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