Calibrating constraint-aware reranking for text-to-SQL execution under 34% schema drift: a threshold audit

ABSTRACT
We evaluated constraint-aware reranking for text-to-SQL execution under 34% schema drift. A deterministic paired simulation generated 56 cases and preserved a boundary-condition stratum. Mean execution accuracy changed from 0.534 to 0.571; the paired difference was +0.036 (95% interval +0.035 to +0.038). 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

Performance averages can obscure whether schema drift changes the reliability of text-to-SQL execution. 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 boundary-condition stratum. The operating setting is low-load; the stress level is fixed at 34% before analysis.

2. Methods

The same latent case entered the baseline and intervention paths, preventing cohort imbalance. We generated 56 cases with seed 50145. 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-R05-145.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 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. For the error slice, [3] supplies abstract-backed evidence on sql, database through the study CIR-SQL: A Dual-Model Intent Recognition Framework for Chinese Text-to-SQL. Its evidence ledger records the specific descriptors backtracking, interference, containing, decoupling, monitoring, operators, frequent, speaking, control, status, leads, seven, sort, representations, classification, hierarchical, intermediate, maintenance; 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 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, [5] 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 measurement, [6] supplies abstract-backed evidence on sql, database through the study Large Language Model Based Chatbot for Database Interaction through Natural Language. Its evidence ledger records the specific descriptors completeness, complexities, naturalness, empowering, interprets, usefulness, interpret, barriers, cohesion, fluency, number, today, back, democratizing, translates, prototype, relevance, informed; 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 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 error slice, [8] 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 reporting rule, [9] 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 baseline, [10] supplies abstract-backed evidence on sql, database, analytics through the study FinSight AI: Coupling a Large Language Model with a Live NoSQL Database for Conversational Financial Analytics and Rule-Assisted Fraud Screening. Its evidence ledger records the specific descriptors authentication, explanations, observations, transactions, aggregates, dashboards, explaining, heuristics, dashboard, portfolio, recipient, threshold, coupling, finsight, grounded, incoming, supplies, account; 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 threshold audit 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, low-load setting, and parameter table; it does not claim observed field measurements. This keeps the schema drift experiment inexpensive while preserving a rerunnable threshold audit.

4. Results

The baseline mean was 0.534; the intervention mean was 0.571. The paired change was +0.036, with a 95% interval from +0.035 to +0.038. In the prespecified adverse slice, the change was +0.032.

The machine-readable artifact `experiments/01-R05-145.json` contains all 56 paired records for text-to-SQL execution. Consequently, the +0.036 result can be recomputed directly under the low-load setting rather than inferred from prose.

5. Discussion

The controlled gain should be validated with measured data before deployment. For text-to-SQL execution under low-load, 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 threshold audit 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 threshold audit 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 low-load 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. 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. 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

3. Wang, Y., Lv, H., & Qian, Y. (2026). CIR-SQL: A Dual-Model Intent Recognition Framework for Chinese Text-to-SQL. AI, 7(3), 91. https://doi.org/10.3390/ai7030091

4. 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

5. 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

6. Souto Prego, B., Vilares, D., & Cabado Lousa, B. (2024). Large Language Model Based Chatbot for Database Interaction through Natural Language. VII Congreso XoveTIC: impulsando el talento científico, 335-342. https://doi.org/10.17979/spudc.9788497498913.47

7. 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

8. 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

9. 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

10. Rehman, A. U. (2026). FinSight AI: Coupling a Large Language Model with a Live NoSQL Database for Conversational Financial Analytics and Rule-Assisted Fraud Screening. https://doi.org/10.2139/ssrn.7093099