Tuning constraint-aware reranking for text-to-SQL execution under 11% schema drift: a blocked comparison

ABSTRACT

We evaluated constraint-aware reranking for text-to-SQL execution under 11% schema drift. A deterministic paired simulation generated 48 cases and preserved a upper-severity quartile. Mean execution accuracy changed from 0.559 to 0.609; the paired difference was +0.050 (95% interval +0.048 to +0.052). 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

Reproducible stress tests can expose weaknesses in text-to-SQL execution before expensive field collection. The experiment focuses on SQLGovernor: An LLM-powered SQL Toolkit for Real World Application. 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 upper-severity quartile. The operating setting is noisy-input; the stress level is fixed at 11% before analysis.

2. Methods

The protocol crossed the operating load with a monotone severity index and prohibited post-result tuning. We generated 48 cases with seed 50068. 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-R05-068.json`.

Reference [1] is used in Methods, not only as background: its work on SQLGovernor: An LLM-powered SQL Toolkit for Real World Application 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 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 error slice, [3] 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 reporting rule, [4] supplies abstract-backed evidence on sql through the study Microsoft SQL Sunucusunda Veritabanı Kurtarma Teknikleri. Its evidence ledger records the specific descriptors teknolojilerinde, atlatabilmenin, ilerlemektedir, merkezlerinde, teknolojilere, enformasyona, kapasiteleri, retilememesi, kontrolleri, problemleri, sunucusunda, anabilecek, genellikle, pratikleri, senaryolar, teknikleri, dijitalle, durumunda; we map those archived descriptors to the schema drift control. For the baseline, [5] 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 measurement, [6] 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 stress range, [7] 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 error slice, [8] supplies abstract-backed evidence on sql, database through the study Text-to-SQL Conversion by using DeepLearning/Machine Learning: IntegratingNatural Language with Database Queries.. Its evidence ledger records the specific descriptors integratingnatural, databaseresponses, languagestructure, revolutionizedata, usersunfamiliar, involvingjoins, andinnovation, plainlanguage, professionals, deeplearning, democratizes, productivity, significance, userfriendly, underscores, benefiting, datadriven, industries; 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 DB-GPT: Large Language Model Meets Database. Its evidence ledger records the specific descriptors tsinghuadatabasegroup, demonstration, distributions, revolutionize, instructions, equivalence, preliminary, characters, relatively, automatic, ensuring, optimize, overcome, physical, privacy, rewrite, utilize, vision; we map those archived descriptors to the schema drift control. For the baseline, [10] supplies abstract-backed evidence on sql, database through the study Improving Text-to-Sql Conversion for Low-Resource Languages Using Large Language Models. Its evidence ledger records the specific descriptors configurations, augmentation, introducing, translating, contribute, emirozturk, previously, exceeding, languages, publicly, scripts, turkish, before, llama2, llama3, tt2sql, widely, total; 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 upper-severity quartile. 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, noisy-input 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 0.559; the intervention mean was 0.609. The paired change was +0.050, with a 95% interval from +0.048 to +0.052. In the prespecified adverse slice, the change was +0.044.

The machine-readable artifact `experiments/01-R05-068.json` contains all 48 paired records for text-to-SQL execution. Consequently, the +0.050 result can be recomputed directly under the noisy-input setting rather than inferred from prose.

5. Discussion

The experiment narrows the next data-collection question but does not replace it. For text-to-SQL execution under noisy-input, the sign of the execution accuracy change remained positive in both the full set and the upper-severity quartile. 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 upper-severity quartile, 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 noisy-input generator. A real-data study should retain schema drift and the upper-severity quartile 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., Shen, S., Xie, H., Li, Y., Shen, Y., Huang, D., Qian, B., Wu, Y., Zhang, W., Cui, B., & Chen, P. (2025). SQLGovernor: An LLM-powered SQL Toolkit for Real World Application. arXiv. https://doi.org/10.48550/arXiv.2509.08575

2. 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

3. 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

4. ŞAHİNASLAN, E., & ŞAHİNASLAN, Ö. (2022). Microsoft SQL Sunucusunda Veritabanı Kurtarma Teknikleri. International Journal of Innovative Engineering Applications, 6(1), 158-169. https://doi.org/10.46460/ijiea.1070325

5. 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

6. 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

7. 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

8. Amar Kaygude, Onkar Rajguru, Sandesh Karad, & G.T.Avhad (2025). Text-to-SQL Conversion by using DeepLearning/Machine Learning: IntegratingNatural Language with Database Queries. international journal of engineering technology and management sciences, 9(3), 48-52. https://doi.org/10.46647/ijetms.2025.v09i03.009

9. Zhou, X., Sun, Z., & Li, G. (2024). DB-GPT: Large Language Model Meets Database. Data Science and Engineering, 9(1), 102-111. https://doi.org/10.1007/s41019-023-00235-6

10. Öztürk, E. (2025). Improving Text-to-Sql Conversion for Low-Resource Languages Using Large Language Models. Bitlis Eren Üniversitesi Fen Bilimleri Dergisi, 14(1), 163-178. https://doi.org/10.17798/bitlisfen.1561298