Agent Bain vs. Agent McKinsey: A New Text-to-SQL Benchmark for the Business Domain

Anonymous ACL submission

Abstract

In the business domain, where data-driven decision making is crucial, text-to-SQL is fundamental for easy natural language access to structured data. While recent LLMs have achieved strong performance in code generation, existing text-to-SQL benchmarks remain focused on factual retrieval of past records. We introduce CORGI, a new benchmark specifically designed for real-world business contexts. CORGI is composed of synthetic databases inspired by enterprises such as Doordash, Airbnb, and Lululemon. It provides questions across four increasingly complex categories of business queries: descriptive, explanatory, predictive, and recommendational. This challenge calls for causal reasoning, temporal forecasting, and strategic recommendation, reflecting multi-level and multi-step agentic intelligence. We find that LLM performance drops on high-level questions, struggling to make accurate predictions and offer actionable plans. Based on execution success rate, the CORGI benchmark is about 21% more difficult than the BIRD benchmark. This highlights the gap between popular LLMs and the need for real-world business intelligence. We release a public dataset and evaluation framework, and a website for public submissions.

1 Introduction

Text-to-SQL, the task of translating natural language questions (NLQ) into executable SQL queries, has long been recognized as a crucial problem in natural language processing and data management (Zhong et al., 2017; Yu et al., 2018a). Its importance extends far beyond academic interest: in business practice, SQL remains the lingua franca of data science and business intelligence (BI), powering critical workflows in analytics, reporting, and decision-making (Provost and Fawcett, 2013; Davenport and Ronanki, 2018). Many modern enterprises employ data analysts responsible for querying structured databases (Davenport, 2006; McAfee and Brynjolfsson, 2012). At consulting firms like McKinsey, SQL and data analysis are also considered essential skills (Bughin et al., 2018).

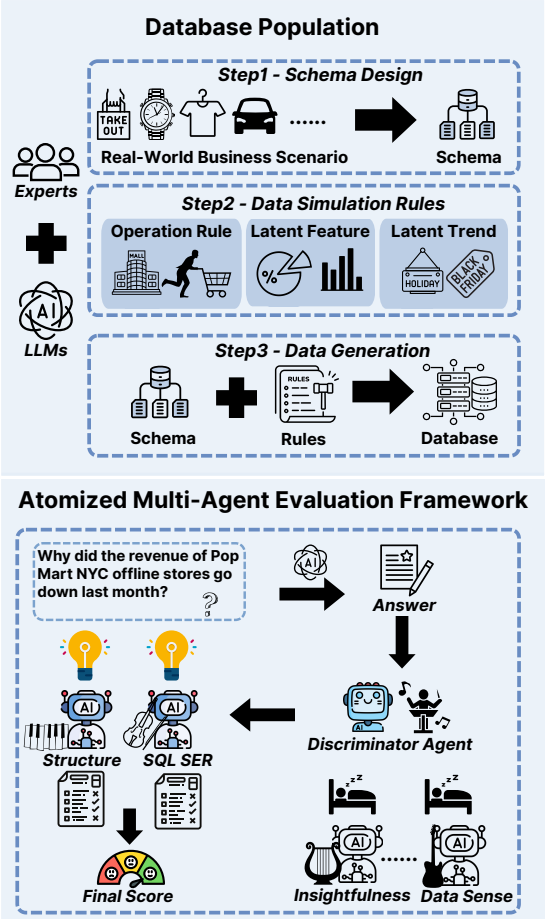


Figure 1: We incorporate business logic into our synthesized database population process, and propose a multi-agent evaluation framework consisting of a discriminator agent and seven scoring agents, each responsible for evaluating one aspect of the generated answer.

Despite LLMs' advances in code generation, existing text-to-SQL benchmarks still fall short of capturing the complexity of real-world BI scenarios. Leading datasets such as BIRD23-train-filtered (Li et al., 2023b), LiveSQLBench (Team,

2025), WikiSQL (Zhong et al., 2017), and Spider2.0 (Team, 2023), primarily emphasize data retrieval rather than enterprise-level reasoning. Nevertheless, decision-makers not only query historical records but also require trend analysis, forecasting, and strategic recommendations (Shmueli, 2010; Fildes et al., 2008; Armstrong, 2001; Davenport and Ronanki, 2018).

To address this gap, we make the following contributions in this paper:

First, we present **CORGI**, a business domain-specific benchmark. To balance realism with open access, CORGI provides simulated databases from verticals such as food delivery, rental, and advertising, as shown in Table 1. CORGI also provides **four categories of business queries**: descriptive, explanatory, predictive, and recommendational, which not only captures real-world retrieval practices but also enables the evaluation of high-level decision-making and strategic planning. In contrast, advanced reasoning questions make up only 0.26% of the BIRD benchmarks' dev set.

Second, we introduce a **business literature inspired evaluation framework** and two new evaluation mechanisms - an **atomized multi-agent evaluation mechanism** and ***CORGI Online* platform**. The atomized multi-agent evaluation mechanism is shown in Figure 1. Based on the nature of business queries, the discriminator agent selects relevant scoring metrics and forwards them to seven specialized scoring agents. This ensures scalability and contextual precision.

Third, with preliminary experiments, we show that leading LLMs exhibit **significant performance drops** on explanatory, predictive and recommendational tasks, highlighting open challenges for future research.

We release the CORGI dataset, evaluation questions, and the evaluation framework on GitHub,[1] along with a public submission platform.[2] These resources are intended to encourage community participation and facilitate further research.

2 Related Work

2.1 Toward Business Intelligence: Expanding the Scope of Text-to-SQL

The BIRD benchmark family has established itself as a cornerstone in the text-to-SQL task. The BIRD team[3] has progressively developed a comprehensive ecosystem since 2023. The original BIRD benchmark (Li et al., 2023b) introduced more than 12k NLQ–SQL pairs across 95 large-scale databases spanning 37 domains, setting a new standard for cross-domain evaluation. This foundation is expanded with BIRD Mini-Dev (2024), a lightweight subset for rapid prototyping; BIRD-CRITIC (2024–2025), extending beyond query generation to SQL debugging and critique across multiple dialects; BIRD-Interact (2025), which models conversational and agentic interactions; and LiveSQLBench (2025), designed for evaluation on evolving and realistic database environments.

Despite these advances, commercial BI scenarios remain largely underexplored. Prior works underscore the strategic value of prescriptive queries for decision-making (Lo et al., 2023) and show that giving actionable recommendations is central to enterprise practice (Gathani et al., 2022). Nevertheless, SQL queries coding proficiency alone is insufficient for BI (Al Maruf et al., 2022).

2.2 Holistic Evaluation Framework.

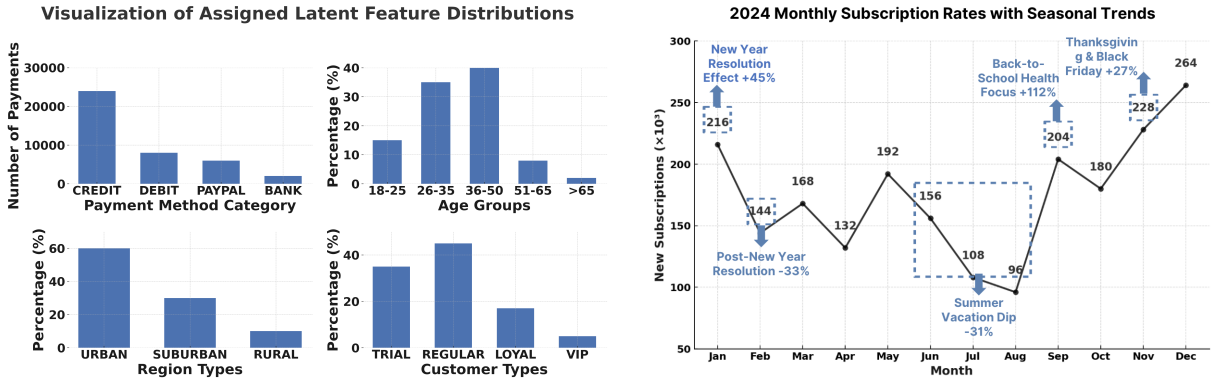
Traditional text-to-SQL evaluations emphasize the correctness of SQL queries and their execution results relying on metrics such as exact match or execution accuracy (Rajpurkar et al., 2016; Liang et al., 2017; Wang et al., 2019; Xu et al., 2021). But these metrics are unable to capture the multi-faceted reasoning and judgment required in real-world decision-making (Bommasani et al., 2021; Zhou et al., 2023).

Recent research in management consulting contexts begin to explore this broader evaluative need. Malberg et al. (2025) and Perga (2024) assess whether LLMs can serve as effective management consultants, pointing out the limitations of LLMs in judgment under uncertainty. Complementary perspectives emphasize that evaluating LLMs in business requires strategic alignment and creativity (Khaldy and Gheraibia, 2025). New studies also stress balancing innovation with ethical and efficiency concerns (Jain, 2024).

3 Benchmark Construction

The dataset construction process of CORGI follows a multi-stage pipeline designed to balance realism

[1] github.com/corgibenchmark/CORGI
[2] txt2sql.com
[3] BIRD team has been based at the University of Hong Kong since 2023, in collaboration with academic and industry partners such as Google Cloud. More details are available at bird-bench.github.io/.



(a) Latent feature distribution (partial)

(b) Latent seasonal trends

Figure 2: Illustrations of data simulation rules without compounding effects, using the *Persona Nutrition* database as a case study.

with the scalability and sharability of synthetic data generation.

3.1 Database Population

Step 1: Construct a schema. We collect information from publicly available websites, applications, and technical documents to extract domain-specific workflows and business logic across diverse verticals. Based on this information, we construct entity–relationship diagrams (ERD) and schemas that approximate the operational structures of real-world business.

Step 2: Simulate data To capture complexities beyond static schemas and to approximate realistic commercial scenarios, we introduce three types of simulation rules to guide data generation. These rules are modeled in isolation as a simplifying assumption; no compounding effects are considered. This serves as the baseline for our study.[4] For example, payment method distributions are not linked to seasonal trends: during Black Friday simulations, the proportions of payment methods are generated randomly rather than conditioned on sales surge. To illustrate the rules **a,b,c**, we simulate *Persona Nutrition*, a personalized health and wellness subscription service, as a case study.

(a) Business Operation Rules. These are deterministic rules that directly reflect the company's operational logic. At *Persona Nutrition*, new customers must complete (i) an *Initial Assessment*. If the health assessment score falls below 3.0, (ii) a *Low Rating Trigger* automatically generates consultation offers. Finally, (iii) *Follow-up Assessments* are conducted every six months or upon user request. These operational constraints are kept consistent with the underlying schemas.

(b) Latent Feature Distribution. Demographic and behavioral patterns are instantiated through predefined distributions. For example, figure 2a illustrates that at *Persona Nutrition*, the customer base consists of 35% trial users (0–3 months, price-sensitive), 44% regular subscribers (3–12 months, consistent users), 17% loyal customers (1–3 years, high engagement), and 4% VIP customers (3+ years, premium services).

(c) Latent Seasonal Trends. Our data also mimic realistic fluctuations. Figure 2b illustrates these seasonal dynamics at *Persona Nutrition*: subscriptions increase during events such as New Year's resolutions (+45%) and Black Friday promotions (+27%), while decreases are observed during periods such as post-New Year resolution (–33%) and summer vacation (–31%).

We employ Gemini-2.5-flash-lite[5] (DeepMind, 2025) to synthesize data consistent with defined schemas and simulation rules. To ensure fidelity, all generated data is manually reviewed and randomly tested. An example prompt for data generation is provided in Appendix B.

3.2 Question Generation

Business data queries can generally be divided into two broad categories: those concerning the *past* and those oriented toward the *future* (Shmueli, 2010; Fildes et al., 2008). Building on this distinction, we categorize our queries into 4 types, as shown in Figure 3. We employ Gemini-2.5-

[4]Future work will model interactions across rules.

[5]Version date: Sept 2025

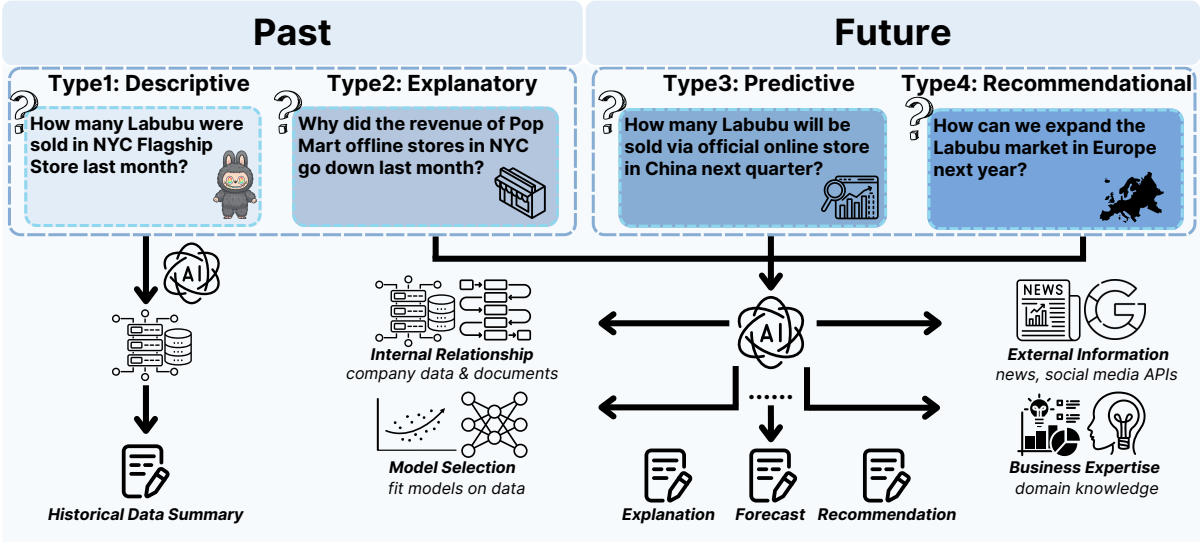


Figure 3: Business questions range in complexity from simple fact retrieval to speculative business strategy.

flash-lite to generate questions for each type. All questions are then manually reviewed and refined.

Type 1: Past, Descriptive. Queries are primarily factual retrieval corresponding to the classical text-to-SQL task (Zhong et al., 2017; Yu et al., 2018a).

For example, "What were the total Labubu sales on September 1, 2025?" can be answered by generating SQL queries (e.g., `SELECT SUM(sale_amount) FROM sales WHERE product_name = 'Labubu' AND sale_date = '2025-09-01';`) to aggregate Labubu transaction data from database.

Type 2: Past, Explanatory. The response requires explanatory reasoning and must interpret historical patterns rather than merely extracting past data (Shmueli, 2010). This demands advanced reasoning capabilities, as LLMs must identify underlying relational patterns and provide plausible causal explanations.

For example, "Why did the revenue of Pop Mart offline stores in New York City (NYC) go down in the last 90 days?" can be answered by attributing the decline to external competition (e.g., the opening of new Jellycat NYC store) or to internal factors such as supply chain disruptions and inventory shortages.

Type 3: Future, Predictive. Predictive queries involve forecasting future outcomes based on historical data (Armstrong, 2001; Fildes et al., 2008). This requires LLMs to uncover trends in past data, formulate assumptions, and select appropriate predictive models (e.g., time series analysis (Faw et al., 2025), linear regression, and LSTM (Hochreiter and Schmidhuber, 1997)). In more complex cases, multiple features may be integrated to produce robust forecasts.

For instance, "What is the expected number of Labubu sales through the North America Online Flagship Store next month?" requires combining historical sales trajectories with contextual signals. Short-term forecasting may incorporate upcoming promotional events such as Black Friday, aligned with historical demand patterns, and modeled through multivariate regression or an LSTM.

Type 4: Future, Recommendational. Queries go beyond description, explanation, and prediction by requiring actionable recommendations that inform decision-making (Davenport and Ronanki, 2018; Provost and Fawcett, 2013). This requires LLMs to synthesize evidence from multiple sources, evaluate alternative strategies, and propose interventions aligned with business objectives.

For instance, consider the query: "How can the Labubu market in Europe be expanded next year?" A prescriptive solution may involve recommending multi-channel expansion strategies, such as: (1) launching region-specific marketing campaigns that leverage local cultural trends, (2) forming strategic partnerships with popular retailers.

3.3 CORGI Statistics

CORGI is derived from 10 representative real-world business verticals. As shown in Table 1, these verticals span three broad categories: consumer platforms (e.g., food delivery, home rental,

Business Scenario	Representative Archetype	Business Model	Tables	#C/T	#R
Food delivery	Doordash	B2C/B2B2C	30	4.0	25
Home rental	Airbnb	C2C	18	4.3	25
Clothing e-commerce	Lululemon	B2C	34	3.9	40
Freelancing marketplace	Upwork	C2C/C2B	25	3.8	33
App store	Google Play	B2C/B2B2C	18	4.3	22
Car rental	Turo	C2C	30	6.7	68
Luxury consignment	The RealReal	C2C	30	6.5	65
User analytics	Ux_ad	B2B	20	7.5	25
Personalized product	Persona Nutrition	B2C	28	7.8	38
E-commerce enablement	Shopify	B2B	23	7.0	60

Table 1: Databases included in the CORGI benchmark, along with their corresponding business models and statistics. #C/T and #R denote the average number of columns per table and the number of foreign key relationships, respectively.

freelancing), retail and commerce (e.g., clothing, luxury consignment, e-commerce enablement), and digital services (e.g., online advertising, app stores, personalized products, user analytics). Detailed descriptions of each scenario are provided in Appendix A. Table 2 compares CORGI with other benchmarks. On average, CORGI contains 26 tables per database, which is significantly more than benchmarks such as BIRD (7.3). Moreover, CORGI is the only benchmark that supports BI questions spanning multiple domain-specific data sources.

Dataset	#D	#T/DB	Domain	BI Questions
Spider[1]	138	5.1	Cross	×
WikiSQL[2]	–	1	Cross	×
BIRD[3]	37	7.3	Cross	×
Advising[4]	1	10	Single	×
CORGI	10	26	Cross	✓

Table 2: CORGI vs. other benchmarks. #D and #T/DB denote the number of databases and the average number of tables per database, respectively.

4 Evaluation Framework

4.1 Motivation

Business management consulting is a multi-trillion dollar industry, central to business and MBA programs and a professional network of firms such as McKinsey & Co.,[6] Boston Consulting Group,[7] and Bain & Company.[8] There is extensive literature on traditional consulting methods offering guidance on strategic reasoning and recommendations (Christensen, 2013; McKinsey & Company, 2020). The recent rise of LLM-based agents increasingly used for decision-making and advice-giving (Park et al., 2023; Yao et al., 2023) sheds new light on these approaches. Bridging these core values with new technology, we survey traditional criteria for evaluating professional consultants and recreate a grading blueprint for assessing LLM agents as consultants.

4.2 Strategic Management Consulting–Inspired Evaluation Framework

With close advising from business school faculty, professional consultants, and corporate practitioners, we develop a strategic management consulting-based framework, grounded in established consulting literature and MBA textbooks (Greiner and Poulfelt, 2005; Kuhn and Hanke, 2010; McKinsey & Company, 2020; Ellet, 2023).

The framework is organized into seven main dimensions:[9] (1) Structure (Minto, 2009), (2) SQL Success Execution Rate (SER), (3) Data Sense (Binder and Watkins, 2024; Chambers et al., 1971), (4) Insightfulness (Backler et al., 2025; Binder and Watkins, 2024), (5) Operational Implementability (Garner, 2013; Chambers et al., 1971), (6) Purpose Alignment (Porter and Kramer, 2011), and (7) Compliance (Guo et al., 2025; Cowgill et al., 2020). Each dimension is further elaborated through submetrics. The full framework description is provided in Appendix C.

Some metrics are universally applicable across type 2, 3 and 4 questions, while others are tailored

[1] Spider (Yu et al., 2018b)
[2] WikiSQL (Zhong et al., 2017)
[3] BIRD (Li et al., 2023b)
[4] Advising (Finegan-Dollak et al., 2018)
[6] mckinsey.com/
[7] bcg.com/
[8] bain.com/

[9] The framework is applied to type 2, 3, and 4 questions.

to specific question type. For example, *Structure* and *SQL SER* serve as requirements for all types. All answers are expected to have a clear logical organization and coherent argumentation, as well as accurate information. However, *Compliance* is only relevant to type 4 questions, as risk management, regulatory compliance, and ethical responsibility are essential considerations when providing recommendations.

4.3 The Combination of Automatic Evaluation and Human Evaluation

Our evaluation system offers both automatic and manual human assessments to balance scalability with judgmental depth. Automatic evaluation offers efficiency across large-scale submissions, while human evaluation provides nuanced interpretation of reasoning quality and domain appropriateness. For each new submission, the system computes its automatic evaluation score available immediately and periodically features it in randomized human comparison rounds. The harmonic averaged score is featured after sufficient human evaluation has been collected.

4.4 Atomized Multi-Agent Evaluation Mechanism

Our multi-agent evaluation mechanism comprises of one discriminator agent and seven scoring agents that collaborate in an orchestrated manner. Consistent with the evaluation framework, *Structure* and *SQL SER* agents serve as general scoring agents (applied to all question types) , while other agents function as specialized scoring agents (applied to certain question type).

The overall mechanism is illustrated in Figure 1, and the evaluation process proceeds as follows: **(1) Input**: Each evaluation instance consists of three component: question type, question, and corresponding answer. **(2) Discriminator step**: The discriminator agent determines relevant scoring dimensions of questions based on both their types and answers. **(3) Scoring step**: two general scoring agents and five specialized scoring agents jointly assign scores to the answers. **(4) Aggregation**: The final score for a given question is obtained by averaging the outputs of all dimension-specific scoring agents involved. The discriminator's role in dynamically selecting relevant scoring agents ensures contextual precision (Fan et al., 2024). By allowing agents to specialize in different evaluation dimensions, this architecture mirrors human committee-based assessment, where diverse perspectives enhance reliability and validity (Chan et al., 2023; Rasheed et al., 2024).

The atomized multi-agent evaluation mechanism is significantly better aligned with human evaluation than the straight-forward mechanism using a single LLM. Figure 4 shows the comparison. Hotelling's T^2 test obtains the following results

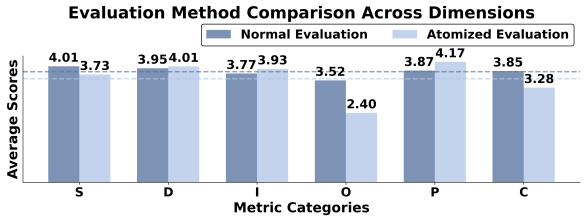


Figure 4: Comparison between our proposed atomized multi-agent evaluation mechanism and single LLM evaluation. S: Structure, D: Data Sense, I: Insightfulness, O: Operation Implementability, P: Purpose Alignment, C: Compliance. The dashed lines indicate overall average scores.

with 6 and 83 degrees of freedom for the numerator and denominator: T^2 statistic is 160.51, F statistic is 25.23 ($p < 0.0001$), indicating that the two evaluation mechanisms produce significantly different outcomes.

4.5 CORGI Online: A Platform for Human Evaluation

We propose *CORGI Online*[10], a web platform for collecting human feedback through pairwise comparisons. Preferences are modeled using the Bradley–Terry framework (Bradley and Terry, 1952). The implementation details of this framework are in Appendix E. This process supports statistically robust aggregation of human judgments for evaluating system outputs.

5 Methodology

5.1 Problem Formulation

Given an NLQ Q, a database schema D, and a knowledge set $K = \{K_I, K_E\}$, where K_I denotes internal knowledge (e.g., a business description of the Shopify database provided by the user) and K_E denotes external knowledge (e.g., information retrieved from web search APIs), a model parameterized by θ consumes prompts P to generate corresponding queries and answers. The system returns an answer A, a list of SQL queries S, and their

[10] `txt2sql.com`

Algorithm 1: Query Generation and Answer Synthesis for a Single NLQ

Initialize $S \leftarrow \emptyset$, $R \leftarrow \emptyset$
$S \leftarrow h(Q, D, K, P \mid \theta)$
for $s \in S$ **do**
  $r \leftarrow e(s)$
  Append r to R
$A \leftarrow f(Q, R, K \mid \theta)$
return A, S, R

execution results R. The complete procedure for generating responses to an NLQ using LLMs is presented in Algorithm 1 and illustrated as follows:

Query generation. $h(\cdot|\theta)$ denotes the query generator which maps (Q, D, K, P_i) to a list of SQL queries S:

$$S = h(Q, D, K, P|\theta).$$

Query execution. $e(\cdot)$ denotes the executor where results R are obtained by running queries on the database defined by D:

$$R = e(S).$$

Answer composition. An answer composer $f(\cdot|\theta)$ produces the final answer from the question and the collected evidence:

$$A = f(Q, R, K|\theta).$$

Output. The system returns an answer A with provenance (S, R). The lists are index-aligned, i.e., each pair (s_i, r_i) corresponds to a query and its execution result. Erroneous queries are discarded or replaced with empty results while maintaining the alignment.

5.2 Scoring Strategy

CORGI adopts a structured two-stage scoring strategy that integrates dimension-specific evaluation with scenario-sensitive aggregation.

Step 1: Dimension Scoring. Each answer is evaluated across the relevant dimensions identified by the discriminator agent, using a scoring scale from 0 to 5 for general criteria, while *SQL SER* is assessed based on a percentage score that reflects execution success. A score of 5 indicates excellent performance in the corresponding dimension, meeting or exceeding professional consulting standards, whereas a score of 0 represents a critical failure.

Step 2: Score Aggregation. Scores are averaged across all evaluation dimensions. The final score of an NLQ answer is computed as:

$$\text{FinalScore} = \frac{1}{n} \sum_{i=1}^{n} \text{Dimension Score}_i,$$

where n denotes the number of evaluation metrics applicable to the given NLQ.

6 Experiments

6.1 Experimental Setup

To kick off the benchmark, we experiment with several popular and cost-effective API-accessible models. For answer generation, we employ *Gemini-2.0-flash-lite* (Google DeepMind, 2024), *Gemini-2.5-flash-lite* (DeepMind, 2025), and *GPT-4o* (OpenAI, 2024). For automatic evaluation, we use *Gemini-2.5-flash-lite* and *GPT-4o*. To ensure reproducibility (Lei et al., 2024; Li et al., 2023a), we set the *temperature* parameter to 0. To maintain the independence of candidate outputs and mitigate sampling bias as well as positional noise (Wang et al., 2023), we adopt a *Single-Answer Evaluation* setting. The prompts used for the atomized evaluation are provided in Appendix B.

6.2 Experimental Results and Discussion

We examine LLMs' performances on higher-level BI questions (types 2, 3, and 4) and observe the following:

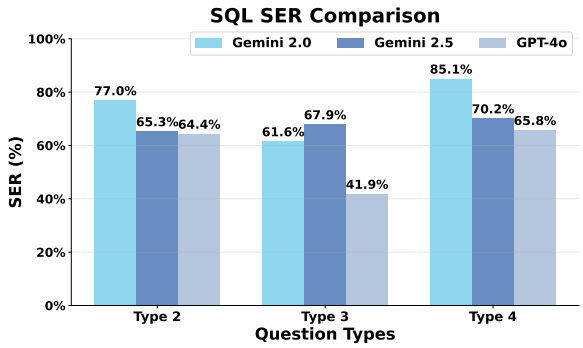


Figure 5: LLMs SQL Query Execution Performance Comparison Across Question Types.

Finding 1: LLMs exhibit a low SQL execution success rate on higher-level questions. As shown in Figure 5, the average SQL SERs of the evaluated LLMs for type 2, type 3, and type 4 questions are 67.8%, 57.4%, and 74.5%, respectively. For the BIRD benchmark, the average SQL SER

Type	Model	S		D		I		O		P		C		Overall	
		G2.5	G4o	G2.5	G4o	G2.5	G4o	G2.5	G4o	G2.5	G4o	G2.5	G4o	G2.5	G4o
Type 2	Gemini 2.0	3.36	3.81	2.96	4.12	2.85	3.59	—	—	—	—	—	—	**3.06**	**3.84**
	Gemini 2.5	3.41	3.87	2.90	4.10	2.92	3.62	—	—	—	—	—	—	**3.08**	**3.86**
	GPT-4o	3.56	3.89	3.45	3.85	3.12	3.60	—	—	—	—	—	—	**3.38**	**3.78**
Type 3	Gemini 2.0	3.38	3.66	2.82	3.71	2.96	3.51	—	—	—	—	—	—	**3.05**	**3.63**
	Gemini 2.5	3.40	3.73	2.87	3.77	3.03	3.56	—	—	—	—	—	—	**3.10**	**3.69**
	GPT-4o	3.49	3.82	3.27	3.68	3.19	3.61	—	—	—	—	—	—	**3.32**	**3.70**
Type 4	Gemini 2.0	3.52	3.88	3.42	4.18	2.55	2.86	1.82	2.48	3.46	4.17	3.53	3.46	**3.05**	**3.51**
	Gemini 2.5	3.56	3.95	3.37	4.13	2.59	2.88	1.87	2.53	3.50	4.23	3.56	3.51	**3.08**	**3.58**
	GPT-4o	3.66	3.99	3.70	4.10	2.70	3.03	2.18	2.49	3.79	4.14	3.60	3.71	**3.27**	**3.54**

Table 3: Comprehensive comparison of different models across evaluation dimensions and overall scores. S: Structure, D: Data Sense, I: Insightfulness, O: Operational Implementability, P: Purpose Alignment, C: Compliance. Dashes (—) indicate metrics not applicable for that question type. The last two columns (**Overall**) report aggregated scores corresponding to Gemini 2.5 (G2.5) and GPT-4o (G4o) evaluators.

reaches 88.0%, which is 21.4% higher than that of CORGI, indicating that CORGI poses substantially greater challenges compared to BIRD.

Finding 2: LLMs differ in evaluating identical answers. As shown in Table 3, GPT-4o consistently assigns higher scores than Gemini 2.5, with an average difference of 0.56 points ($p < 0.0001$), suggesting that Gemini 2.5 serves as a more stringent evaluator. However, neither model exhibits self-preference, i.e., a systematic tendency to assign higher scores to its own responses. For instance, on type 2 questions, Gemini 2.5 assigns an average score of 3.08 to its own outputs (compared to an overall mean of 3.17), whereas GPT-4o assigns 3.78 to its own answers (compared to an overall mean of 3.83).

Finding 3: LLMs yield progressively lower scores from type 2 to type 4 questions. As shown in Figure 6, average scores decline steadily from type 2 to type 4 questions, indicating degraded LLM performance with higher task complexity. ANOVA confirmed significant differences among conditions (Gemini 2.5: $F = 3.50$, $p = 0.0398$, $p < 0.05$; GPT-4o: $F = 248.56$, $p < 0.001$). The drop is modest but statistically significant for Gemini 2.5, whereas it is substantial and highly significant for GPT-4o.

Finding 4: LLMs struggle to produce actionable plans for type 4 questions. Type 4 responses receive the lowest overall evaluation scores, mainly driven by their low *Operation Implementability* (2.23). Although their *Structure* score is relatively high (3.76), which suggests that while LLMs can organize answers well, they still struggle to generate feasible, solution-oriented action plans comparable to those of professional consultants.

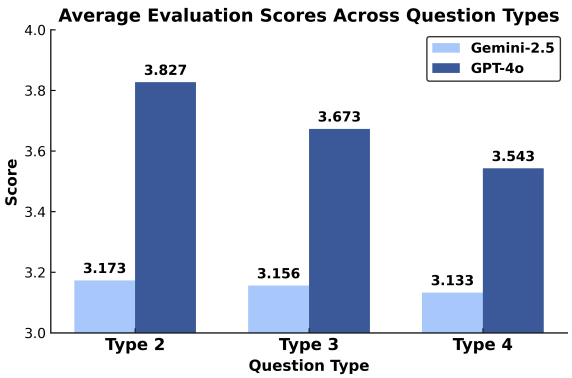


Figure 6: LLMs yield progressively lower scores from type 2 to type 4 questions.

7 Conclusion

We introduce CORGI, a new text-to-SQL benchmark featuring 10 hand-curated industry-inspired databases, a new business literature-informed evaluation framework, and two new evaluation mechanisms. We show that popular LLMs struggle with increasingly difficult business insight questions such as those requiring recommendations. While LLMs are good at generating structured, MBA-trained responses, they fail to give sound implementable action plans. We release CORGI as a public benchmark for researchers to contribute to and help to further our understanding of LLMs as management consultants.

Limitations

While CORGI advances schema realism and task relevance for business applications, several limitations remain. First, all database contents and SQL queries are synthesized rather than drawn from production systems; this ensures privacy and scalability but omits natural artifacts such as noise, redundancy, and schema drift. Second, CORGI focuses on single-turn queries, leaving out multi-turn conversational refinement that is common in real-world systems. Third, each query is paired with only one natural language formulation, limiting the evaluation of model robustness to paraphrasing or informal inputs. These limitations highlight promising directions for extending CORGI in future work.

Ethical Considerations

Ethical considerations are very important in our research. Firstly, our evaluation framework includes an Ethical Responsibility metric, designed to assess whether analyses and recommendations align with established principles of AI ethics. Secondly, no external human labeling or crowd-sourced annotations are involved in our work; all data labeling and quality assurance were conducted exclusively by the authors. Thirdly, we release our dataset, questions, and evaluation framework on GitHub. This allows other researchers to replicate, critique, and extend our findings, contributing to collective progress in the field. Nevertheless, regarding data privacy, all data used in constructing the dataset originate from publicly available sources, with no inclusion of confidential information. By taking these steps, we aim to propose a new SQL evaluation benchmark while ensuring its responsible and equitable implementation.

Acknowledgments

Left blank for anonymous submission.

References

Abdullah Al Maruf, Rajesh Paul, Mohammad Hasan Imam, and Zahir Babar. 2022. A systematic review of the role of sql and excel in data-driven business decision-making for aspiring analysts. *American Journal of Scholarly Research and Innovation*, 1(01):249–269.

J. Scott Armstrong. 2001. *Principles of Forecasting: A Handbook for Researchers and Practitioners*. Springer.

W. Backler, A. Iny, N. D'Intino, E. Parker, and S. Hirashita. 2025. Navigating the future with strategic foresight.

Jonathan Binder and Michael D. Watkins. 2024. To solve a tough problem, reframe it. *Harvard Business Review*, 102(1-2):80–89.

Rishi Bommasani, Drew A Hudson, Ehsan Adeli, Russ Altman, Simran Arora, Sydney von Arx, Michael Bernstein, Jeannette Bohg, Antoine Bosselut, Emma Brunskill, and 1 others. 2021. On the opportunities and risks of foundation models. *arXiv preprint arXiv:2108.07258*.

Ralph Allan Bradley and Milton E. Terry. 1952. Rank analysis of incomplete block designs: I. the method of paired comparisons. *Biometrika*, 39(3/4):324–345.

Jacques Bughin, Jeongmin Seong, James Manyika, Michael Chui, and Raoul Joshi. 2018. Notes from the ai frontier: Applications and value of deep learning. *McKinsey Global Institute*.

J. C. Chambers, S. K. Mullick, and D. D. Smith. 1971. How to choose the right forecasting technique. *Harvard Business Review*.

Chi-Min Chan, Weize Chen, Yusheng Su, Jianxuan Yu, Wei Xue, Shanghang Zhang, Jie Fu, and Zhiyuan Liu. 2023. Chateval: Towards better llm-based evaluators through multi-agent debate. *Preprint*, arXiv:2308.07201.

Laurence J. Christensen. 2013. *The Business of Consulting*. Wiley.

Bo Cowgill, Francesca Dell'Acqua, Senyu Deng, Daniel Hsu, Nitin Verma, and Augustin Chaintreau. 2020. Biased programmers? or biased data? a field experiment in operationalizing ai ethics. In *Proceedings of the 21st ACM Conference on Economics and Computation*, pages 679–681.

Thomas H. Davenport. 2006. Competing on analytics. *Harvard Business Review*.

Thomas H. Davenport and Rajeev Ronanki. 2018. *Artificial Intelligence for the Real World*. Harvard Business Publishing.

Google DeepMind. 2025. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. *Preprint*, arXiv:2507.06261.

William Ellet. 2023. *The Case Study Handbook: A Student's Guide*, 2nd edition. Harvard Business Review Press.

Zhihao Fan, Jialong Tang, Wei Chen, Siyuan Wang, Zhongyu Wei, Jun Xi, Fei Huang, and Jingren Zhou. 2024. Ai hospital: Benchmarking large language models in a multi-agent medical interaction simulator. *Preprint*, arXiv:2402.09742.

Matthew Faw, Rajat Sen, Yichen Zhou, and Abhimanyu Das. 2025. In-context fine-tuning for time-series foundation models. In *Forty-second International Conference on Machine Learning*.

Robert Fildes, Paul Goodwin, Michael Lawrence, and Konstantinos Nikolopoulos. 2008. Forecasting and operational research: A review. *Journal of the Operational Research Society*, 59(9):1150–1172.

Catherine Finegan-Dollak, Jonathan K. Kummerfeld, Li Zhang, Karthik Ramanathan, Sesh Sadasivam, Rui Zhang, and Dragomir R. Radev. 2018. Improving text-to-sql evaluation methodology. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 351–360.

Bryan A. Garner. 2013. *HBR Guide to Better Business Writing*. Harvard Business Review Press.

Mehul Gathani and 1 others. 2022. What-if analysis for business users: Current practices and future opportunities. *arXiv preprint arXiv:2212.13643*.

Google DeepMind. 2024. Introducing gemini 2.0: our new ai model for the agentic era. https://blog.google/technology/google-deepmind/google-gemini-ai-update-december-2024/. Accessed: 2025-10-07.

Larry E Greiner and Flemming Poulfelt. 2005. *Consulting to Management*. Springer.

Jiayu Guo, Yu Guo, Martha Li, and Songtao Tan. 2025. Flame: Financial large-language model assessment and metrics evaluation. *Preprint*, arXiv:2501.06211.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural Computation*, 9(8):1735–1780.

Souratn Jain. 2024. Revolutionizing business consulting with generative ai: Exploring transformative models for strategic decision-making, innovation, and operational excellence. *World Journal of Advanced Research and Reviews*, 22:2273–2284.

Mohammad Al Khaldy and Youcef Gheraibia. 2025. Evaluation and mitigation of the limitations of large language models in business decision-making. In *2025 1st International Conference on Computational Intelligence Approaches and Applications (ICCIAA)*, pages 1–6.

Thomas Kuhn and Friedrich Hanke. 2010. *Management Consulting: A Guide to the Profession*. International Labour Organization.

Fangyu Lei, Jixuan Chen, Yuxiao Ye, Ruisheng Cao, Dongchan Shin, Hongjin Su, Zhaoqing Suo, Hongcheng Gao, Wenjing Hu, Pengcheng Yin, and 1 others. 2024. Spider 2.0: Evaluating language models on real-world enterprise text-to-sql workflows. *arXiv preprint arXiv:2411.07763*.

Jinyang Li, Binyuan Hui, Ge Qu, Jiaxi Yang, Binhua Li, Bowen Li, Bailin Wang, Bowen Qin, Ruiying Geng, Nan Huo, Xuanhe Zhou, Chenhao Ma, Guoliang Li, Kevin C.C. Chang, Fei Huang, Reynold Cheng, and Yongbin Li. 2023a. Can llm already serve as a database interface? a big bench for large-scale database grounded text-to-sqls. In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, NIPS '23, Red Hook, NY, USA. Curran Associates Inc.

Zihan Li, Ruibo Cao, Tao Zhang, Lei Li, and Zhoujun Li. 2023b. Bird: A benchmark for biomedical text-to-sql. In *Advances in Neural Information Processing Systems (NeurIPS)*.

Xiaojun Liang, Yanyan Li, Jian Zhang, and Percy Liang. 2017. Seq2sql: Generating structured queries from natural language using reinforcement learning. In *ICLR*.

Victor S. Y. Lo and 1 others. 2023. From meaningful data science to impactful decisions. *Data Science Journal*, 22(1):1–15.

Simon Malberg, Nikita Grigorev, Victoria Lea Rein, Barbara Borne Bass, Maria Alejandra Gelvez Alvarez, and Georg Groh. 2025. Bridging ai and business: Are large language models good management consultants? In *ECIS 2025 Proceedings*, Artificial Intelligence in Organizations and Society. Paper No. ECIS2025-1108, CRP.

Andrew McAfee and Erik Brynjolfsson. 2012. Big data: The management revolution. *Harvard Business Review*.

McKinsey & Company. 2020. The state of organizations 2020. https://www.mckinsey.com/. Accessed: 2025-09-29.

Barbara Minto. 2009. *The Pyramid Principle: Logic in Writing, Thinking, and Problem Solving*, 3rd edition. Financial Times Prentice Hall, Harlow, UK.

OpenAI. 2024. Gpt-4o system card. *Preprint*, arXiv:2410.21276.

Joon Sung Park, Joseph C O'Brien, Carrie J Cai, Meredith Ringel Morris, Percy Liang, and Michael S Bernstein. 2023. Generative agents: Interactive simulacra of human behavior. *arXiv preprint arXiv:2304.03442*.

Iurii Perga. 2024. The role of ai (large language models) in managerial decision-making: Benefits and challenges. *(Naukovì perspektivi)*.

Michael E. Porter and Mark R. Kramer. 2011. Creating shared value. *Harvard Business Review*, 89(1-2):62–77.

Foster Provost and Tom Fawcett. 2013. *Data Science for Business: What You Need to Know about Data Mining and Data-Analytic Thinking*. O'Reilly Media, Inc.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. Squad: 100,000+ questions for machine comprehension of text. In *EMNLP*.

Zeeshan Rasheed, Muhammad Waseem, Kari Systä, and Pekka Abrahamsson. 2024. Large language model evaluation via multi ai agents: Preliminary results. *Preprint*, arXiv:2404.01023.

Galit Shmueli. 2010. To explain or to predict? *Statistical Science*, 25(3):289–310.

BIRD Team. 2025. Livesqlbench: Benchmarking text-to-sql in evolving and realistic database environments. In *Proceedings of the 2025 Annual Meeting of the Association for Computational Linguistics (ACL)*. https://livesqlbench.ai.

Spider Team. 2023. Spider2.0: Towards comprehensive evaluation of text-to-sql systems. https://yale-lily.github.io/spider2/.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. 2019. Glue: A multi-task benchmark and analysis platform for natural language understanding. In *ICLR*.

Peiyi Wang, Lei Li, Liang Chen, Zefan Cai, Dawei Zhu, Binghuai Lin, Yunbo Cao, Qi Liu, Tianyu Liu, and Zhifang Sui. 2023. Large language models are not fair evaluators. *Preprint*, arXiv:2305.17926.

Pengcheng Xu, Tao Yu, Tianze Shi, Rui Jiang, Chenglong Li, Zhiwei Dong, Tao Zhao, Ruoxi Cao, Jun Yan, Thomas Lin, and 1 others. 2021. Sparc and cosql: Cross-domain context-dependent text-to-sql semantic parsing. In *ACL*.

Shunyu Yao, Dian Yu, Jeffrey Zhao, Izhak Shafran, Thomas L Griffiths, Yuan Cao, and Karthik Narasimhan. 2023. Tree of thoughts: Deliberate problem solving with large language models. In *NeurIPS*.

Tao Yu, Rui Zhang, Kai Yang, Michihiro Yasunaga, Dongxu Wang, Zifan Li, James Ma, Irene Li, Qingning Yao, Shanelle Roman, Zilin Zhang, and Dragomir Radev. 2018a. Spider: A large-scale human-labeled dataset for complex and cross-domain semantic parsing and text-to-sql task. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Tao Yu, Rui Zhang, Kai Yang, Michihiro Yasunaga, Dongxu Wang, Zifan Li, James Ma, Irene Li, Qingning Yao, Shanelle Roman, Zilin Zhang, and Dragomir Radev. 2018b. Spider: A large-scale human-labeled dataset for complex and cross-domain semantic parsing and text-to-sql task. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 3911–3921.

Victor Zhong, Caiming Xiong, and Richard Socher. 2017. Seq2sql: Generating structured queries from natural language using reinforcement learning. *CoRR*, abs/1709.00103.

Xinyi Zhou, Frank F. Xu, and 1 others. 2023. Limitations of current text-to-sql evaluation frameworks. In *Workshop on Data-Centric AI, NeurIPS*.

A Detailed Descriptions of CORGI Business Scenarios

We provide the following additional details for the 10 business scenarios in CORGI to give readers a clearer view of the covered domains.

1 Online food delivery platform (e.g. Doordash) This type of business connects users with a variety of local restaurants, and connects delivery personnel with different orders. The platform differentiates itself through personalized features including allergen tracking, prime club membership, customizable favorite lists, and family account management. Dynamic business features such as time-bound promotions and surge pricing allow restaurants to optimize engagement and profitability. Users can browse restaurants by categories, view detailed menus organized into sections and dishes, and place order. Orders progress through detailed status stages, and are fulfilled by a fleet of delivery personnel tracked by availability and vehicle type, with their performance recorded in a delivery history log. Users can rate not only restaurants but also individual dishes and delivery service, while restaurants can respond directly to reviews. The system supports secure payments, real-time tracking, promotional codes, and supports reviews and replies to ensure feedback loops.

2 App store (e.g. Google Play) This type of business connects developers and users through app discovery, download, and monetization. Users can log in socially, maintain balances, and use multiple payment methods; developers can publish apps with version control, pricing models, and platform support; apps are categorized and tagged for enhanced discoverability; and usage is tracked through downloads, installations, rankings, and reviews.

3 Online clothing store (e.g. Lululemon) This type of business usually operates an online and offline fashion retail platform that offers a diverse range of products categorized by department, category, and collection, and is tailored to various customer needs through uses, features, and fit types. The schema is centered on the Products table, which connects to departments, categories, collections, and is extended via junction tables for product uses and features. Each product links

to one or more SKUs, which represent sellable items with inventory, pricing, and shipment details. Customer data is also extensively modeled, including multiple addresses, payment methods, loyalty membership, and order history, while promotional codes, returns, cancellations, and fulfillment options are designed to support scalable and efficient operations across both digital and physical retail channels.

4 Freelancing platform (e.g. Upwork) This type of business operates a freelancing platform that connects employers and freelancers globally. Users can be differentiated between employers and freelancers. Employers can post jobs, each categorized and requiring specific skills. Freelancers can maintain a detailed profile with portfolio, education, experience and certifications. They can apply for the jobs that employers posted, which may lead to contracts. The wage is handles by different payment methods. After the completion of contracts, employers and freelancers can give reviews to each other.

5 Online room booking platform (e.g. Airbnb) This type of business operates in the short-term rental and vacation housing market, connecting hosts who want to rent out properties with guests seeking accommodation. Users can act as either Hosts or Guests. Listings are tied to specific neighborhoods, cities, and countries through hierarchical foreign keys. Each listing includes details such as RoomTypes, pricing, amenities, availability, and policies. Booking activities are linked to users and listings, and supported by payment infrastructure (Payments, PaymentMethods, UserPaymentMethods). The system also tracks user engagement through Reviews and UserFavorites.

6 Personalized products (e.g. Persona Nutrition) This type of business operates a platform that offers customized products tailored to individual customer needs. Users typically begin by completing a detailed assessment, which is then analyzed to provide personalized recommendations. For example, in the case of nutrition, the platform might offer custom vitamin packs, but this model can extend to other product categories such as skincare, beauty, and wellness products. Each product is linked to specific SKUs, representing items tailored to the customer's preferences. The business model is often subscription-based, ensuring regular delivery of the customized products. Additionally, the platform may offer reordering features, customer support for further customization, and a feedback loop to refine future product recommendations based on user reviews and changing needs.

7 Car Rental services (e.g.Turo) This type of business operates a peer-to-peer rental platform that connects vehicle owners with individuals seeking short-term car rentals. Vehicle owners, known as hosts, can list their cars on the platform, specifying the car's availability, location, and rental price. Renters, known as guests, can browse these listings, select a vehicle, and book it for the desired period. The platform charges a commission on each transaction, with the commission percentage varying based on factors like the type of vehicle and rental duration. Each rental is managed through the system, tracking reservations, pricing, insurance, and payment processing. Additionally, features like user verification, vehicle delivery, and 24/7 support ensure a seamless and secure experience for both parties. The platform also allows renters to rate hosts and vice versa after each transaction.

8 Second-hand luxury goods marketplace (e.g. TheRealReal) This type of business operates a consignment platform specializing in the sale of pre-owned luxury goods, including clothing, handbags, jewelry, and art. Sellers consign their items to the platform, where experts authenticate each item before listing it for sale. The platform earns revenue by taking a commission from each sale, with rates varying based on the category and sale price of the item. The items are categorized and linked to detailed product pages, showcasing the brand, condition, and price. Buyers can browse and purchase authenticated luxury items from a wide range of high-end brands. The platform also offers additional services such as home pickup, personalized styling, and premium membership, enhancing the overall customer experience. Additionally, customers can sell their own pre-owned items through the platform, fostering a circular economy for luxury goods.

9 UX-driven advertising analytics platform (e.g. Mixpanel + Amplitude + AppsFlyer hybrid) This type of business provides a full-stack data platform focused on optimizing digital advertising and customer experience through behavioral analytics, A/B testing, and multi-channel attribution. The platform integrates data from advertisement impressions, clicks, user sessions, content views,

and CRM records to offer a unified view of the customer journey. Advertisers and product teams can launch campaigns, define audience segments, and test multiple ad creatives or landing pages using built-in experimentation tools. The platform supports real-time event tracking and visual funnel analysis, allowing businesses to identify drop-offs, optimize conversion paths, and personalize experiences at scale. The company also enables dynamic cohort building, personalized campaign delivery, and feedback-loop integration with external ad platforms (e.g., Meta, Google), helping brands make more informed, data-driven decisions to maximize ROI and user engagement.

10 E-commerce enablement platform (e.g. Shopify) This type of business provides an end-to-end platform that enables entrepreneurs, SMBs, and large enterprises to launch, operate, and scale online storefronts with minimal technical complexity. The platform offers drag-and-drop website builders, integrated payment processing, inventory and order management, and multi-channel sales capabilities across marketplaces, social media, and in-person POS systems. It integrates data from product catalogs, customer interactions, and transaction histories to support a seamless retail experience. Merchants can run marketing campaigns, personalize product recommendations, and automate workflows using built-in analytics and third-party apps. The platform supports real-time inventory syncing, flexible fulfillment options, and automated tax and shipping calculations. Businesses can enhance customer engagement through CRM tools, loyalty programs, and abandoned cart recovery.

B Prompts

B.1 Data Simulation Rules Generation

The following is the prompt for generating data simulation rules, using Shopify as an example.

Prompt for Generating Data Simulation Rules

INSTRUCTIONS: Assume you are a data scientist working on a text-to-SQL benchmark project. You now need to generate data simulation rules for Shopify database. To better approximate real business scenario of Shopify, you have: (1) Shopify description; (2) Link to Shopify official website; (3) Schema (***shopify.sql***); (4) Real-world user experience report. You can retrieve information from reliable online sources.

OBJECTS: Your need to design simulation rules three categories of data simulation rules: (1) Business Operation Rules: Deterministic rules that directly encode Shopify's operational logic; (2) Latent Feature Distributions: Probabilistic rules capturing demographic and behavioral regularities; (3) Latent Trends: Temporal dynamics that reflect realistic fluctuations (e.g., seasonal demand shifts).

REQUIREMENTS: All rules should be documented in ***rules.md***, clearly delineated by category. When constructing the rules, adhere strictly to the tables, columns, and relationships explicitly defined in ***shopify.sql***. Do not introduce assumptions about missing or inferred schema elements.

B.2 Database Population

The following is the prompt for populating database, using Shopify as an example.

Prompt for Populating Database

INSTRUCTIONS: Assume you are a data scientist working on a text-to-SQL benchmark project. You now need to populate the Shopify database. You are provded with: ((1) Schema (***shopify.sql***); (2) Data simulation rules (***rules.md***).

OBJECTS: You should populate the Shopify database by strictly following ***shopify.sql*** and ***rules.md***. You should populate the database as realistic as possible, simulating the real e-commerce business enablement platform scenario.

REQUIREMENTS: The database should be stored in ***shopify.db***.

B.3 Answer Generation

The following are the prompts for answer generation.

Prompt for SQL Queries Generation

System Prompt:
You are an expert SQL analyst. Given a database schema and a business question, generate multiple comprehensive SQL queries (up to {max_queries}) that will help answer the question from different angles.
Guidelines:
1. Generate at most {max_queries} different SQL queries that explore the question comprehensively.
2. Return each query in a separate code block with a brief description.
Format your response as:
Query 1: [Brief description]

```
SELECT ...
FROM ...
WHERE ...
```

Query 2: [Brief description]

```
SELECT ...
FROM ...
WHERE ...
```

And so on.
User Prompt:
Database Schema: {schema}
Question: {question}
Generate multiple SQL queries to comprehensively answer this question:

Prompt for Final Answer Generation

You are a senior data analyst. Given a business question and multiple SQL queries with their results, provide a comprehensive analysis that synthesizes all the information.
Original Question: {question}
All Queries and Results: {query_list}, {result_list}

B.4 Evaluation

The following is the prompt for metric evaluation.

Prompt for Metric Evaluation

You are an expert evaluator. Please evaluate the following answer based on the given question using the {metric} metric.
Question: {question}
Answer: {answer}
Metric Definition: {metric}: {metric definition}
Evaluation Criteria: {criteria}
Instructions: Please provide your evaluation in the following JSON format:
{{
"Score": <score 0-5>,
"Reasoning": "<brief explanation of the score>"
}}
Please be precise and provide clear reasoning for your score. The output format should be strictly as above. Do not include any other text other than the JSON format.

C Complete Strategic Management Consulting-based Evaluation Framework

1. Structure
1.1 Argument Soundness
Coverage: Type 2, 3, 4
Definition: To assess whether the subpoints of answer are Mutually Exclusive and Collectively Exhaustive (MECE), a core part of the Minto Pyramid Principle.
1.2 Logical Coherence
Coverage: Type 2, 3, 4
Definition: (i) Contextual alignment: To assess whether the answer aligns with the context of the question (e.g., timeframe, entities). (ii) Reasoning alignment: To assess whether the chain of reasoning is complete and sound.
1.3 Verbosity
Coverage: Type 2, 3, 4
Definition: To assess whether the answer is concise and not overly detailed.

2. SQL Success Execution Rate
Coverage: Type 2, 3, 4
Definition: To measure the proportion of generated SQL queries that can be executed successfully without syntax or runtime errors.

3. Data Sense
3.1 Information Adequacy
Coverage: Type 2, 3, 4
Definition: To assess whether internal or/and external information included in the answer is adequate to answer the question.
3.2 Trend Awareness
Coverage: Type 2, 3, 4

Definition: To assess whether the external and internal trends are explored in the answer.

3.3 Model Selection Rationale

Coverage: Type 3 - numerical prediction

Definition: To assess whether an rational model is chosen to make predictions.

4. Insightfulness

4.1 Out-of-box Thinking

Coverage: Type 2, 3, 4

Definition: To assess if the analysis introduces conceptual advancements or solutions outside common analytical parameters.

4.2 Root Cause Depth

Coverage: Type 2

Definition: To assess whether the analysis moves beyond surface-level symptoms to identify plausible, well-supported root causes.

4.3 Assumption Appropriateness

Coverage: Type 3

Definition: To assess (i) whether the assumption is appropriate and realistic, (ii) whether the results match the assumptions.

5. Operation Implementability

5.1 Actionability

Coverage: Type 4

Definition: To assess whether the recommendation is executable and concrete.

5.2 Time-Based Planning

Coverage: Type 4

Definition: To assess whether the recommendation includes specific implementation timeline.

6. Purpose Alignment

6.1 Goal Orientation

Coverage: Type 4

Definition: To assess whether the recommendation is associated with reachable and motivational goals.

6.2 Stakeholder Orientation

Coverage: Type 4

Definition: To assess if the recommendation explicitly considers and balances the interests and impacts on key non-shareholder stakeholders (e.g., employees, customers, community).

7. Compliance

7.1 Risk Management

Coverage: Type 4

Definition: To assess the inclusion of adequate risk warnings, identifications of potential risk points, and provision of risk mitigation suggestions.

7.2 Regulatory Compliance

Coverage: Type 4

Definition: To assess adherence to laws, regulations, and industry standards.

7.3 Ethical Responsibility

Coverage: Type 4

Definition: To assess if the analysis and recommendations uphold principles of AI ethics (e.g., fairness, well-being, non-discrimination), moving beyond mere legal adherence.

D Evaluation Examples

Type 2: Why did download numbers in March 2025 much higher than other months? (Appstore db)

Responder: Gemini-2.5-flash-lite

The data reveals a significant spike in app downloads in March 2025, with **3006 downloads** compared to an average of approximately **1800–1900 downloads** in surrounding months (Query 1). This analysis investigates the contributing factors behind this surge.

Key Drivers of the Download Increase:

- **Dominance of "BlackMonkey" App:** The primary driver of the download surge in March 2025 was overwhelmingly the app **"BlackMonkey"**, which accounted for **1222 downloads** (Query 2). This single app represents a substantial portion of the total downloads for that month, far exceeding any other individual app.
- **Developer "KidsCloud" Success:** The developer **"KidsCloud"** appears to be closely linked to this surge, reporting **1208 downloads** in March 2025 (Query 7). Given that "BlackMonkey" is the top-downloaded app and "KidsCloud" is the top-downloaded developer, it is highly probable that "BlackMonkey" is developed by "KidsCloud" and is the reason for their exceptional download numbers.
- **Category Performance:** The surge is concentrated within the **"Sports"** (1364 downloads) and **"Food &**

Drink" (1251 downloads) categories (Query 4). This suggests that "BlackMonkey" (or any other significant contributors from KidsCloud) likely falls into one or both of these categories.

- **Age Group Engagement:** The "Teen" (**1824 downloads**) and "Kids" (**1820 downloads**) age groups were the most active in March 2025 (Query 5). This aligns with the hypothesis that "BlackMonkey" is an app popular with younger demographics, potentially a game or an entertainment app.
- **Platform Distribution:** While iOS and Android show high download numbers (**2307** and **2299** respectively), macOS and Windows also saw a considerable increase in downloads compared to other months (Query 3). This indicates that the surge in downloads for "BlackMonkey" (and potentially other apps from KidsCloud) was not limited to a single platform.
- **Pricing Model:** Both **"paid"** (**1284 downloads**) and **"free"** (**1200 downloads**) apps saw a significant increase in downloads, with subscription models also showing a moderate rise (**522 downloads**) (Query 6). This suggests that the increased downloads might be a mix of paid app popularity and free app adoption, or that "BlackMonkey" itself might have a tiered pricing model or in-app purchases.

App Releases and Updates in March 2025: Query 8 lists several app releases and updates in March 2025. While many popular apps like PicsArt, Nike Training Club, and Clash of Clans had updates, none of these individually appear to have the download volume to explain the overall surge. The app **"BlackMonkey"** does not appear in the list of updated or released apps in March 2025. This suggests that either "BlackMonkey" had no significant update or release in March, or that the surge is driven by an app not captured in this specific query (e.g., an app with a very long name not easily identifiable, or an app released prior to March but gaining significant traction due to external factors).

Conclusion: The significant increase in app downloads in March 2025 was primarily driven by the app **"BlackMonkey"**, likely developed by **"KidsCloud"**. This app appears to be particularly popular among **teen and kid** demographics and has likely contributed heavily to the **Sports** and **Food & Drink** categories. The download surge was observed across major platforms (iOS, Android, macOS, Windows) and was influenced by both paid and free app downloads. Further investigation could focus on understanding the specific marketing efforts or viral nature of "BlackMonkey" that led to this exceptional performance.

Evaluator: GPT-4o

Scores:

- Structure
  - Argument Soundness: 5
  - Logical Coherence: 5
  - Verbosity: 3
- SQL SER: 1
- Data Sense
  - Information Adequacy: 5
  - Trend Awareness: 3
- Insightfulness
  - Out-of-box Thinking: 3
  - Root Cause Depth: 5

Type 3: What will our return rate be during the next holiday season? (Clothing E-commerce DB)

Responder: GPT-4o

Comprehensive Analysis. The original question seeks to estimate the return rate for the next holiday season based on historical data. By synthesizing the results of the provided queries, we can derive insights into the overall return rate, the primary reasons for returns, the categories with the highest

return rates, and the impact of customer loyalty tiers on returns. These insights will help us predict and potentially mitigate returns during the next holiday season.

1. Overall Return Rate (Query 1)

The overall return rate during the last holiday season (November 1, 2022, to December 31, 2022) was **11.52%**. Out of 382 total orders, 44 were returned. This provides a baseline for estimating the return rate for the upcoming holiday season. If similar conditions persist (e.g., product offerings, customer base, and operational factors), we can expect a return rate in the range of **11–12%**.

2. Primary Reasons for Returns (Query 2)

The analysis of return reasons reveals the following key insights:

- The top reasons for returns were **"Didn't fit" (26.47%)**, **"Wrong size" (23.53%)**, and **"Quality issue" (23.53%)**. Together, these three reasons accounted for over 73% of all returns.
- Other reasons, such as **"Defective item" (14.71%)** and **"Changed mind" (11.76%)**, were less frequent but still notable.

This suggests that sizing and quality are the primary drivers of returns. To reduce the return rate, the company could:

- Improve size guides and provide more detailed product descriptions.
- Enhance quality control processes to minimize defective items.
- Offer virtual fitting tools or customer reviews to help customers make more informed decisions.

3. Return Rates by Product Category (Query 3)

The return rates varied significantly across product categories:

- **Dresses (16.96%)**, **Swimwear (15.70%)**, and **Shoes (13.97%)** had the highest return rates. These categories are likely more prone to sizing and fit issues, which aligns with the reasons for returns identified in Query 2.
- Categories like **Bottoms (8.45%)** and **Formal Wear (10.00%)** had lower return rates.

This indicates that the company should focus on addressing fit-related concerns in high-return categories like Dresses, Swimwear, and Shoes. For example:

- Offering free exchanges for different sizes in these categories could reduce the likelihood of outright returns.
- Providing detailed size charts and customer reviews specific to these categories could also help.

4. Impact of Customer Loyalty Tiers on Returns (Query 4)

The return rates by loyalty tier were as follows:

- **Silver (14.47%)** and **Gold (14.12%)** customers had the highest return rates.
- **Platinum (7.50%)** customers had the lowest return rate, followed by **Bronze (10.39%)** and customers with no loyalty tier (10.94%).

This suggests that higher-tier loyalty customers (Silver and Gold) are more likely to return items, possibly because they purchase more frequently or feel more confident in the return process. Platinum customers, on the other hand, may be more selective in their purchases, leading to fewer returns.
To address this:

- The company could analyze the purchasing behavior of Silver and Gold customers to identify patterns leading to higher returns.
- Offering personalized recommendations or targeted size/fit guidance for these tiers could help reduce returns.

5. Key Takeaways and Recommendations

Based on the analysis, the following insights and actions can help predict and mitigate the return rate for the next holiday season.

Predicted Return Rate. The return rate for the next holiday season is likely to be in the range of **11–12%**, consistent with the last holiday season.

Recommendations to Reduce Returns.

1. **Address Sizing and Fit Issues:**
   - Focus on high-return categories like Dresses, Swimwear, and Shoes by improving size guides, offering virtual fitting tools, and providing detailed product descriptions.
   - Consider offering free exchanges for size-related issues to reduce outright returns.
2. **Enhance Quality Control:**
   - Since "Quality issue" accounted for 23.53% of returns, improving quality checks and addressing common defects could significantly reduce returns.
3. **Targeted Interventions for Loyalty Tiers:**
   - Analyze the purchasing behavior of Silver and Gold customers to identify patterns leading to higher returns.
   - Provide personalized recommendations or targeted size/fit guidance for these tiers.
4. **Monitor and Address Seasonal Trends:**
   - Since the holiday season often involves gifting, returns due to "Changed mind" or "Wrong size" may increase. Offering gift receipts or extended return windows could improve customer satisfaction while managing returns.

Evaluator: Gemini-2.5-flash-lite
Scores:

- Structure
  - Argument Soundness: 5
  - Logical Coherence: 5
  - Verbosity: 0
- SQL SER: 0.9
- Data Sense
  - Information Adequacy: 3
  - Trend Awareness: 3
  - Model Selection Rationale: 0
- Insightfulness
  - Out-of-box Thinking: 0
  - Assumption Appropriateness: 3
  - Root Cause Depth: 5

Type 4: How can I increase the repeat purchase rate? (Shopify DB)

Responder: GPT-4o

Comprehensive Analysis. Based on the query results, there is no data for customers with repeat purchases (i.e., customers with more than one order). This indicates that all existing customers have made only a single purchase. The absence of repeat orders may result from several factors, including limited post-purchase engagement, lack of incentives to return, or misaligned marketing strategies. Below is an analysis of the findings and actionable recommendations to improve the repeat purchase rate.

1. Key Observations

(a) No Repeat Purchases. Query 1 confirms that there are no customers with more than one order (`total_orders = 1`). Subsequent queries (2–8) analyzing patterns such as discounts, campaigns, delivery, or refunds also return no results, since no repeat customers exist.

(b) Marketing Campaigns. Despite substantial ad spending (e.g., up to $24,156.55), Query 5 shows that marketing campaigns have not generated repeat customers, sug-

gesting a weak alignment between marketing efforts and retention goals.
(c) Fulfillment and Delivery. Query 6 reports no delivery delays for repeat customers; however, this finding is irrelevant given the absence of such customers.
(d) Discounts and Promotions. Query 4 reveals no data on discounts driving repeat purchases, implying that either discount-based campaigns are lacking or ineffective.
(e) Product Offerings. Query 3 indicates that no specific product variant or collection has encouraged repeat purchases.
(f) Refunds. Query 8 shows no refund data related to repeat customers—unsurprising given their absence—suggesting that refund issues are not the primary barrier to repeat purchases.

2. Recommendations to Increase Repeat Purchases

1. Understand Customer Behavior (Root-Cause Analysis).

- Conduct post-purchase surveys or interviews to identify reasons customers do not return.
- Evaluate satisfaction with product quality, pricing, and service experience.
- Verify that product value propositions align with customer expectations.

2. Enhance Customer Experience.

- **Personalization:** Use purchase history and preferences to recommend relevant products and send follow-up emails.
- **Onboarding:** Deliver a seamless first-purchase experience that builds trust and recall.
- **Fulfillment:** Offer fast and reliable delivery to reinforce satisfaction.
- **Quality Assurance:** Maintain consistent product quality to encourage repeat buying.

3. Introduce Loyalty Programs.

- Implement a points-based system rewarding repeat purchases.
- Provide tiered benefits (e.g., Bronze, Silver, Gold) to incentivize engagement.

4. Run Retargeting and Re-engagement Campaigns.

- Re-engage one-time customers via email, SMS, or social media.
- Highlight new arrivals, exclusive offers, and limited-time discounts.
- Use abandoned-cart reminders to recover potential repeat buyers.

5. Leverage Discounts and Incentives.

- Offer second-purchase discounts (e.g., "10% off your next order").
- Provide free shipping or gifts for repeat customers.
- Introduce bundles or subscriptions to foster habitual purchases.

6. Improve Marketing Effectiveness.

- Shift focus from acquisition to retention-oriented marketing.
- Target existing customers with retargeting ads and personalized messaging.
- Evaluate the ROI of current channels (affiliate, social) and expand to email or partnership marketing.

7. Expand Product Offerings.

- Introduce complementary or add-on products to encourage follow-up purchases.
- Bundle products to increase perceived value and convenience.

8. Build Trust and Brand Loyalty.

- Communicate brand values and share authentic customer stories.
- Feature reviews or user-generated content to build credibility.
- Engage actively with customers on social platforms.

9. Monitor and Optimize.

- Track metrics such as repeat customer rate, average order value, and customer lifetime value.
- Continuously experiment with retention tactics and optimize based on results.

3. Immediate Next Steps

1. **Survey existing customers** to collect feedback on satisfaction and barriers to repeat purchases.
2. **Launch a "Welcome Back" campaign** offering targeted discounts to first-time buyers.
3. **Analyze marketing ROI** and reallocate budget toward retention activities.
4. **Design a loyalty program** to reward repeat behavior.

Evaluator: GPT-4o
Scores:

- Structure
  - Argument Soundness: 5
  - Logical Coherence: 5
  - Verbosity: 3
- SQL SER: 0.3
- Data Sense
  - Information Adequacy: 5
  - Trend Awareness: 3
  - Model Selection Rationale: 3
- Insightfulness
  - Out-of-box Thinking: 5
  - Assumption Appropriateness: 3
- Operation Implementability
  - Actionability: 5
  - Time-based Planning: 0
- Purpose Alignment
  - Goal Orientation: 5
  - Stakeholder Orientation: 5
- Compliance
  - Risk Management: 5
  - Regulatory Compliance: 3
  - Ethical Responsibility: 5

E Implementation Details of Bradley–Terry Framework

We use the Bradley–Terry framework to model pairwise preferences and implement it with the `choix` PyPI package (v0.3.6). We assign each item i a latent strength parameter $\theta_i > 0$. For n comparisons, we uniformly sample a pair (i, j) from all $\binom{n}{2}$ possible pairs and determine the outcome through a Bernoulli trial with $P(i \succ j) = \frac{\theta_i}{\theta_i + \theta_j}$. We record (a, b) when a is preferred over b, producing a consistent dataset for ranking evaluation under stochastic preferences.