Avaliação Sistemática de Text-to-SQL em Português: Um Benchmark Unificado Multi-Domínio

Guilherme Fonseca[1], Julio C. S. Reis[2], Matheus Botelho[1], Pedro Calais[1], Emerson A. Simoes[1], Lucas R. Silva[1], Eduardo Camara[3], Jaide Zardin[3], Rodrigo Gonçalves[1], Gustavo Ribeiro[1], Yago Lobato[3], Allan Sene[4], Altigran S. Silva[3], Marcos A. Gonçalves[1]

[1] Universidade Federal de Minas Gerais (UFMG),
[2] Universidade Federal de Viçosa (UFV)
[3] Universidade Federal do Amazonas (UFAM),
[4] Dadosfera Brasil

{guilhermefonseca, matheusbotelho, emerson.simoes}@dcc.ufmg.br, jreis@ufv.br
{pedrolgcalais, lucasrsilvak, rodrigopfmg, gustavo9661}@ufmg.br
{eduardo.camara, jaide.zardin, yagobrlobato, alti}@icomp.ufam.edu.br
allan@dadosfera.ai, mgoncalv@dcc.ufmg.br

Abstract. *The Text-to-SQL task has progressed significantly with the emergence of Large Language Models (LLMs), but its evaluation in Portuguese remains limited. This work proposes a unified benchmark that combines four public databases from different domains and evaluates six distinct LLMs using Execution Accuracy (EX) and Component Matching (CM) within a standardized experimental protocol. The results indicate that, although models achieve over 70% EX on Spider, the best performance on DATASUS is only 27.5%. The discrepancy between CM and EX also highlights limitations in compositional generalization. These findings suggest that isolated evaluations may overestimate model capabilities and underscore the need for integrated, multi-domain benchmarks.*

Resumo. *A tarefa Text-to-SQL tem avançado significativamente com o uso de Grandes Modelos de Linguagem (LLMs), mas sua avaliação em português ainda é limitada. Este trabalho propõe um benchmark unificado que consolida quatro bases públicas de diferentes domínios e avalia seis LLMs distintos a partir das métricas EX e CM sob um protocolo experimental padronizado. Os resultados mostram que, embora os modelos superem 70% de EX no Spider, o melhor desempenho no DATASUS alcança apenas 27,5%. A discrepância entre CM e EX também evidencia limitações de generalização composicional. Esses resultados sugerem que avaliações isoladas podem superestimar a capacidade dos modelos e destacam a necessidade de benchmarks integrados e multi-domínio.*

1. Introdução

A tarefa de Text-to-SQL tem avançado significativamente com o uso de Grandes Modelos de Linguagem (LLMs, do inglês *Large Language Models*), permitindo a conversão de perguntas em linguagem natural em consultas SQL executáveis [Yu et al. 2018,

Lei et al. 2025]. Apesar desse progresso, a avaliação desses sistemas permanece fortemente concentrada no idioma inglês e baseada em *benchmarks* isolados, como Spider e BIRD [Li et al. 2023, Dou et al. 2023]. Embora amplamente utilizados, esses *benchmarks* frequentemente não refletem a diversidade e a complexidade de cenários reais, podendo levar a estimativas excessivamente otimistas do desempenho dos modelos.

No contexto do português, essa limitação é ainda mais pronunciada. Trabalhos existentes concentram-se em bases específicas ou domínios isolados [Jose and Cozman 2023, Pedroso et al. 2025, Yamate et al. 2025, Moraes et al. 2025], dificultando comparações consistentes e comprometendo a avaliação da capacidade de generalização dos modelos. Mais criticamente, argumentamos que o principal gargalo não é apenas a escassez de dados, mas a *fragmentação dos protocolos experimentais*: a ausência de avaliações integradas que considerem múltiplas bases, modelos e níveis de complexidade sob um mesmo padrão impede uma compreensão mais fiel do desempenho real dos sistemas.

Neste trabalho, propomos uma avaliação sistemática e unificada da tarefa de Text-to-SQL em português, baseada na consolidação de múltiplas bases de dados e na comparação de diferentes LLMs sob um protocolo experimental padronizado. Diferentemente de trabalhos anteriores, este *benchmark* unifica (i) múltiplos domínios reais e sintéticos, (ii) um protocolo experimental homogêneo entre bases e (iii) uma avaliação comparativa *cross-domain* sob condições controladas. Especificamente, nosso objetivo é responder às seguintes questões de pesquisa (QPs): **QP1)** *quais bases estão disponíveis e quais suas características estruturais e semânticas?* e **QP2)** *qual é a efetividade dos LLMs em diferentes cenários e níveis de complexidade?*

Como primeira contribuição, consolidamos um *benchmark* unificado composto por quatro bases públicas — Spider-pt (dev e test), text2SQL4PM e DATASUS (SIH/-SUS) — abrangendo diferentes domínios, estruturas e níveis de dificuldade. Como segunda contribuição, conduzimos uma avaliação sistemática de seis LLMs de código aberto, incluindo modelos generalistas e especializados em código, utilizando métricas complementares (acurácia de execução – EX e correspondência de componentes – CM) sob um protocolo reprodutível e controlado. Até onde temos conhecimento, este é o primeiro trabalho a avaliar múltiplas bases de Text-to-SQL em português sob um protocolo unificado, permitindo comparações diretas e controladas entre diferentes domínios.

Em resumo, nossos resultados revelam um ***gap* substancial entre *benchmarks* tradicionais e cenários reais**. Enquanto modelos alcançam mais de 70% de acurácia de execução no Spider, o melhor desempenho no DATASUS não ultrapassa 27.5%, evidenciando limitações significativas de generalização e robustez semântica. Esse resultado sugere que avaliações baseadas em *benchmarks* isolados podem superestimar de forma sistemática a capacidade dos modelos em contextos reais.

O restante deste artigo está organizado da seguinte forma. A Seção 2 discute os trabalhos relacionados. A Seção 3 descreve o processo de identificação e seleção dos conjuntos de dados. A Seção 4 apresenta a metodologia adotada, incluindo modelos, estratégias de *prompting* e métricas de avaliação. A Seção 5 apresenta e analisa os resultados experimentais. Por fim, a Seção 6 apresenta as conclusões e direções para trabalhos futuros.

2. Trabalhos Relacionados

O desenvolvimento de *benchmarks* é fundamental para o avanço da tarefa Text-to-SQL, pois permite avaliar de forma sistemática a capacidade dos modelos em gerar consultas SQL a partir de linguagem natural [Gao et al. 2024]. Em inglês, *benchmarks* como Spider [Yu et al. 2018] e BIRD [Li et al. 2023] estabeleceram protocolos padronizados que viabilizam comparações consistentes entre modelos e níveis de complexidade.

Em contraste, no contexto do português, o ecossistema permanece fragmentado. Embora existam iniciativas multilíngues [Pham et al. 2025], a avaliação de sistemas Text-to-SQL em português ainda carece de diversidade de bases e, principalmente, de padronização experimental, dificultando comparações diretas entre estudos.

Uma linha de trabalho adapta *benchmarks* consolidados para o português, principalmente via tradução do Spider. José e Cozman [José and Cozman 2021, Jose and Cozman 2023] traduziram as partições de treino e desenvolvimento, enquanto Pedroso et al. [Pedroso et al. 2025] disponibilizaram a partição de teste. Esses esforços permitiram estudos comparativos, como [de Carvalho et al. 2025], que mostram desempenho consistentemente inferior em português e ganhos associados a modelos maiores e treinamento multilíngue. No entanto, essas abordagens herdam limitações do próprio Spider, como baixa representatividade de cenários reais e menor complexidade estrutural [Lei et al. 2025], além de frequentemente adotarem protocolos experimentais distintos.

Em paralelo, outros trabalhos exploram bases mais próximas de cenários reais, como DATASUS [Moraes et al. 2025], Funarte [Petrola and Franco 2025], censo educacional [Fróes and Braghetto 2025] e dados do Instituto Nacional de Estudos e Pesquisas (INEP) [Centenaro 2025]. Embora avancem em realismo, essas iniciativas são tipicamente restritas a domínios específicos e avaliadas de forma isolada, sem integração entre bases, modelos e protocolos.

Dessa forma, a literatura atual se organiza em duas vertentes complementares, porém desconectadas: (i) *benchmarks* traduzidos, que favorecem comparabilidade mas carecem de realismo, e (ii) bases de domínio específico, que aumentam a representatividade mas comprometem a comparabilidade. A ausência de uma avaliação integrada que concilie esses aspectos limita a compreensão do desempenho real dos modelos. Este trabalho endereça essa lacuna ao consolidar múltiplas bases de diferentes domínios em um *benchmark* unificado e ao avaliar, sob um protocolo padronizado, diferentes LLMs, permitindo uma análise comparativa mais abrangente e realista.

3. Processo de Identificação e Seleção das Bases de Dados

A partir de uma revisão da literatura, identificamos e selecionamos bases utilizadas em estudos prévios, adotando como critérios: disponibilidade pública dos dados e anotações SQL, diversidade de domínio e viabilidade experimental. Bases como Funarte [Petrola and Franco 2025], Censo Educacional [Fróes and Braghetto 2025] e INEP [Centenaro 2025] foram excluídas por não estarem publicamente disponíveis para *download* no momento da realização dos experimentos. Como resultado e resposta à nossa **QP1**: *Quais bases de dados estão disponíveis e quais suas características estruturais e semânticas?*, consolidamos e analisamos quatro conjuntos de dados:

Spider-pt (*dev*[1] e *test*[2]) [Jose and Cozman 2023, Pedroso et al. 2025], text2SQL4PM [Yamate et al. 2025][3] e DATASUS (SIH/SUS) [Moraes et al. 2025][4].

3.1. Bases de Dados

A Tabela 3.1 apresenta uma visão comparativa das bases selecionadas, incluindo domínio, volume de pares pergunta-SQL, número de bancos de dados, número de tabelas por banco de dados e número de palavras por pergunta em linguagem natural.

Base de Dados	Contexto	# Pares NL-SQL	# BDs	# Méd. tab.	# Méd. pal.
Spider-pt (dev)	Tradução do fold dev do Spider 1.0	1034	20	4,00 ±2,50	14,04 ± 4,93
Spider-pt (test)	Tradução do fold test do Spider 1.0	2147	40	4,50 ± 2,88	14,41 ± 4,91
text2SQL4PM	Registros de pagamentos e requerimentos	1655	1	6,00 ±0,0	15,84 ± 6,98
DATASUS (SIH/SUS)	Dados de hospitalizações	51	1	15,00 ±0,00	16,25 ± 13,75

Tabela 1. Bases de dados de Text-to-SQL português.

As bases apresentam diferenças relevantes em termos de representatividade e complexidade. O Spider-pt corresponde a versões traduzidas de um *benchmark* amplamente utilizado, preservando sua estrutura original. Em contraste, text2SQL4PM e DATASUS refletem cenários mais próximos de aplicações reais, com maior especificidade de domínio. Destaca-se que o DATASUS é o único conjunto de dados inteiramente em português, enquanto os demais combinam perguntas em português com esquemas ou conteúdos em inglês. Essa escolha pode favorecer modelos com maior exposição ao inglês, potencialmente subestimando desafios específicos da língua portuguesa.

A Figura 1 ilustra a distribuição dos níveis de dificuldade das consultas em cada base de dados, evidenciando variações na complexidade entre os cenários considerados.

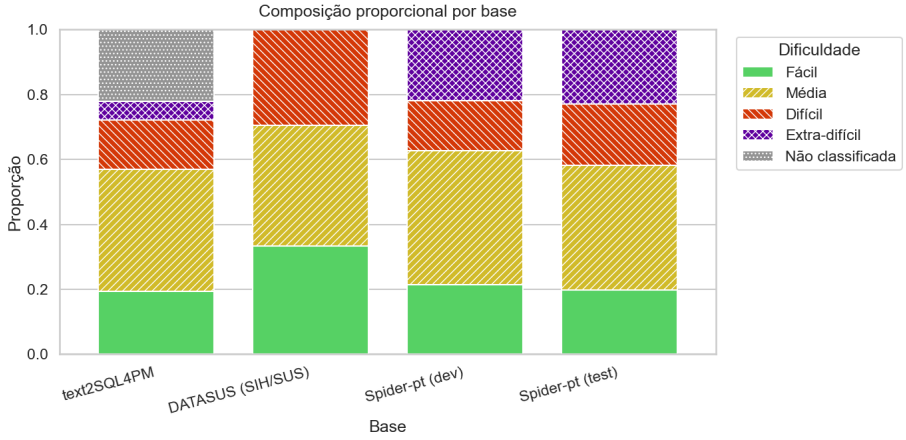

Figura 1. Distribuição dos níveis de dificuldade por base de dados.

[1]https://huggingface.co/datasets/Marchanjo/spider-pt
[2]https://huggingface.co/datasets/Boakpe/spider-test-portuguese
[3]https://github.com/pm-usp/text-2-sql
[4]https://github.com/DAVINTLAB/DATASUSAnalytics

As consultas são classificadas em quatro níveis de dificuldade (fácil, média, difícil e extra-difícil), com base na complexidade estrutural das queries, incluindo número de operações, junções e subconsultas [Yu et al. 2018, Yamate et al. 2025]. No caso do text2SQL4PM, há ainda uma classe adicional ("Não classificada") para consultas com construções não contempladas no critério original do Spider. A Tabela 2 apresenta um exemplo aleatório de consulta para cada nível de dificuldade.

Por fim, observamos heterogeneidade significativa nos protocolos experimentais adotados nos trabalhos originais. Enquanto Spider-pt e text2SQL4PM utilizam configurações mais simples, o DATASUS explora abordagens mais complexas, incluindo estratégias de *prompting* e agentes. Além disso, há variação no conjunto de modelos avaliados em cada estudo, o que reforça a dificuldade de comparação direta entre resultados, motivando a necessidade de uma avaliação unificada.

Dificuldade	**Pergunta**	**SQL**
Fácil	Qual o número de carros com mais de 4 cilindros?	SELECT COUNT(*) FROM cars_data WHERE cylinders > 4
Média	Para cada estádio, quantos concertos existem?	SELECT T2.name, COUNT(*) FROM concert AS T1 JOIN stadium AS T2 ON T1.stadium_id = T2.stadium_id GROUP BY T1.stadium_id
Difícil	Quantos países da europa possuem pelo menos 3 fabricantes de carros?	SELECT T1.country_name FROM countries AS T1 JOIN continents AS T2 ON T1.continent = T2.cont_id JOIN car_makers AS T3 ON T1.country_id = T3.country WHERE T2.continent = 'Europe' GROUP BY T1.country_name HAVING COUNT(*) >= 3
Extra-difícil	Qual a expectativa de vida média para países onde o inglês não é o idioma oficial?	SELECT AVG(life_expectancy) FROM country WHERE name NOT IN (SELECT T1.name FROM country AS T1 JOIN country_language AS T2 ON T1.code = T2.country_code WHERE T2.language = "English" AND T2.is_official = "T")
Não classificada	Quais eventos possuem custo diferente de nulo?	SELECT * FROM event_log WHERE cost IS NOT NULL

Tabela 2. Exemplos de consultas por nível de dificuldade.

4. Metodologia

Detalhes relativos à metodologia proposta para realização deste estudo, incluindo protocolo experimental e métricas para avaliação comparativa, são apresentados a seguir.

4.1. Bases de Dados e Modelos

Consideramos em nossos experimentos todas as bases de dados do *benchmark* proposto - Spider (dev), Spider (test), DATASUS (SIH/SUS) e text2SQL4PM - descritas na Seção 3.

Para a seleção dos modelos, foram analisados os resultados reportados em cada um dos artigos que propuseram as bases de dados utilizadas [Jose and Cozman 2023, Pedroso et al. 2025, Yamate et al. 2025, Moraes et al. 2025], identificando os três melhores modelos para o português em cada trabalho. Desse conjunto, selecionamos apenas os modelos de código aberto, o que resultou na seguinte lista: Qwen2.5-Coder-7B-Instruct, Qwen2.5-Coder-14B-Instruct e Qwen2.5-Coder-32B-Instruct [Hui et al. 2024]; Qwen2.5-32B-Instruct [Qwen Team 2025]; e Llama-3.1-8B-Instruct [Grattafiori et al. 2024]. Para aumentar a abrangência da avaliação, incluímos também o llama-3-sqlcoder-8b[5], um modelo *fine-tunado* especificamente para a tarefa de Text-to-SQL. Ao todo, foram avaliados 6 modelos de LLMs, constituindo uma seleção diversificada que cobre diferentes famílias, escalas de parâmetros e perfis de especialização. Essa seleção busca equilibrar desempenho reportado, diversidade de arquiteturas e escalas, e disponibilidade de modelos de código aberto.

Todos os modelos foram carregados com quantização de 4 bits (NF4) [Dettmers et al. 2023] em bfloat16, por meio da biblioteca BitsAndBytes[6], visando viabilizar a execução em GPUs com diferentes capacidades de memória. A geração das consultas SQL foi realizada utilizando *greedy decoding*, estratégia na qual o modelo seleciona deterministicamente o token de maior probabilidade a cada passo de decodificação [Song et al. 2025]. Essa escolha é motivada por evidências recentes de que o *greedy decoding* supera métodos baseados em amostragem (*sampling*) na maioria das tarefas avaliadas, com impacto particularmente expressivo em tarefas de raciocínio e geração de código [Song et al. 2025]. Além do melhor desempenho, essa estratégia garante a reprodutibilidade total dos resultados, uma vez que a mesma entrada sempre produz a mesma saída. Os experimentos foram conduzidos em máquinas equipadas com GPUs NVIDIA RTX A6000 (48 GB), RTX 4090 (24 GB) e RTX 5090 (32 GB). Como a decodificação é determinística e os modelos são carregados com a mesma configuração de quantização, a variação de hardware não afeta os resultados obtidos.

4.2. Estratégia de *Prompting*

Para todos os modelos, foi adotada uma abordagem *zero-shot*, na qual o modelo recebe apenas o esquema do banco de dados e a pergunta do usuário, sem exemplos de pares pergunta-SQL. A Figura 2 apresenta um exemplo do *prompt* utilizado; para fins de ilustração, apresentamos apenas o esquema de 2 tabelas do banco de dados, porém no cenário real o esquema completo de todas as tabelas é enviado no *prompt*. Optamos por utilizar o *prompt* em inglês, mesmo com perguntas em português, seguindo a prática adotada em trabalhos anteriores [de Carvalho et al. 2025], dado que os LLMs avaliados foram predominantemente pré-treinados em inglês e tendem a responder melhor a instruções nesse idioma.

[5]https://huggingface.co/defog/llama-3-sqlcoder-8b

[6]https://github.com/bitsandbytes-foundation/bitsandbytes

Exemplo de *prompt*

```
[System]:
You are an expert SQL assistant.  Given the database schema below, write a single
valid SQL query that answers the user's question.  Return ONLY the SQL query,
without explanations or markdown.

### Schema
CREATE TABLE admissions (id INTEGER, hospital_id INTEGER, patient_age INTEGER,
diagnosis_code VARCHAR, ...);
CREATE TABLE hospital (id INTEGER, name VARCHAR, city VARCHAR, ...);

[User]:
Quantas internações ocorreram em Porto Alegre em 2020?
```

Figura 2. Exemplo de prompt utilizado na abordagem zero-shot.

4.3. Métricas de Avaliação

A avaliação dos modelos é conduzida por meio de duas métricas complementares, amplamente utilizadas na literatura de Text-to-SQL:

i) **Acurácia de Execução (EX)** [Pedroso et al. 2025], que compara o resultado da execução da consulta SQL gerada pelo modelo com o resultado da consulta de referência (*gold*). Uma predição é considerada correta se ambas as execuções retornam o mesmo conjunto de resultados. Essa métrica é robusta a variações sintáticas válidas, uma vez que consultas SQL diferentes podem produzir o mesmo resultado.

ii) **Correspondência de Componentes (CM)** [Moraes et al. 2025], que avalia a correspondência entre os componentes individuais da consulta gerada e da consulta de referência, considerando cláusulas como SELECT, WHERE, GROUP BY, ORDER BY, entre outras. De forma geral, valores mais altos para ambas as métricas indicam melhor desempenho dos modelos.

5. Resultados Experimentais

Nesta seção, apresentamos os resultados experimentais que respondem à **QP2**: *Qual é a efetividade dos LLMs na tarefa Text-to-SQL em português em diferentes bases e cenários?* Nosso objetivo é analisar não apenas quais modelos apresentam melhor desempenho, mas também como esse desempenho varia em função do domínio e da complexidade das bases.

5.1. Resultados Agregados

A Tabela 3 apresenta os resultados de CM e EX para os seis modelos avaliados nas quatro bases do *benchmark*. Embora a tabela também inclua EX por nível de dificuldade, nesta seção focamos nos valores agregados; a análise por dificuldade é discutida na Seção 5.2.

De forma geral, os modelos da família Qwen2.5 dominam o desempenho, alcançando os maiores valores de CM e EX em todas as bases. Em contraste, os modelos baseados em Llama, incluindo o SQLCoder-8B, apresentam desempenho consistentemente inferior. Entre todos os modelos, o Qwen2.5-Coder-32B se destaca como o mais efetivo, liderando em EX em 3 das 4 bases e em CM em 1 delas.

A comparação entre modelos especializados em código e suas versões generalistas de mesmo porte indica ganhos consistentes decorrentes da especialização. Por exemplo,

modelo	CM	EX	EX				
	Total	Total	Fácil	Média	Difícil	Extra-Difícil	Não Avaliada
DATASUS (SIH/SUS)							
Qwen2.5-Coder-32B-Instruct	57.6%	23.5%	58.8%	**10.5%**	0.0%	-	-
Qwen2.5-Coder-14B-Instruct	**58.7%**	**27.5%**	**64.7%**	**10.5%**	**6.7%**	-	-
Qwen2.5-Coder-7B-Instruct	56.7%	23.5%	**64.7%**	5.3%	0.0%	-	-
Qwen2.5-32B-Instruct	51.1%	19.6%	47.1%	**10.5%**	0.0%	-	-
Llama-3.1-8B-Instruct	50.7%	13.7%	35.3%	5.3%	0.0%	-	-
llama-3-sqlcoder-8b	50.3%	17.6%	52.9%	0.0%	0.0%	-	-
spider (test)							
Qwen2.5-Coder-32B-Instruct	57.1%	**70.7%**	89.4%	**74.5%**	67.9%	**50.3%**	-
Qwen2.5-Coder-14B-Instruct	56.3%	69.7%	**91.3%**	73.3%	65.9%	48.1%	-
Qwen2.5-Coder-7B-Instruct	55.5%	65.2%	87.5%	70.0%	63.5%	39.1%	-
Qwen2.5-32B-Instruct	**58.0%**	68.2%	87.1%	71.2%	**69.1%**	46.0%	-
Llama-3.1-8B-Instruct	54.5%	59.7%	81.4%	64.1%	57.3%	35.7%	-
llama-3-sqlcoder-8b	42.8%	42.5%	69.6%	43.7%	38.8%	20.3%	-
spider (dev)							
Qwen2.5-Coder-32B-Instruct	**59.7%**	**65.6%**	86.5%	69.2%	**64.2%**	**39.2%**	-
Qwen2.5-Coder-14B-Instruct	57.4%	65.3%	86.0%	**69.5%**	62.3%	**39.2%**	-
Qwen2.5-Coder-7B-Instruct	57.8%	59.9%	86.0%	62.9%	60.4%	28.2%	-
Qwen2.5-32B-Instruct	58.1%	63.4%	**87.8%**	67.6%	59.1%	34.8%	-
Llama-3.1-8B-Instruct	55.2%	56.5%	82.0%	58.9%	56.0%	27.3%	-
llama-3-sqlcoder-8b	43.5%	40.8%	71.6%	42.0%	32.7%	14.1%	-
text2SQL4PM							
Qwen2.5-Coder-32B-Instruct	70.6%	**54.4%**	**78.3%**	**53.2%**	49.4%	53.6%	**39.2%**
Qwen2.5-Coder-14B-Instruct	69.7%	54.1%	77.6%	52.3%	55.4%	53.6%	35.9%
Qwen2.5-Coder-7B-Instruct	66.5%	47.6%	66.8%	44.8%	52.6%	52.6%	30.4%
Qwen2.5-32B-Instruct	**72.5%**	54.1%	77.6%	52.9%	61.4%	53.6%	30.4%
Llama-3.1-8B-Instruct	66.0%	53.5%	69.9%	50.0%	**70.1%**	**64.9%**	30.4%
llama-3-sqlcoder-8b	59.4%	35.5%	48.1%	35.2%	37.5%	28.9%	25.2%

Tabela 3. Resultados de CM e EX dos LLMs nos diferentes conjuntos de dados. EX é separado por dificuldade; "-" indica ausência desse nível. Negrito marca o melhor resultado por base.

o Qwen2.5-Coder-32B supera sua versão generalista (Qwen2.5-32B) em EX no Spider (test) (70.7% vs. 68.2%), Spider (dev) (65.6% vs. 63.4%) e no DATASUS (23.5% vs. 19.6%), além de apresentar desempenho semelhante no text2SQL4PM. No entanto, esse comportamento não se repete na família Llama, onde o SQLCoder-8B não supera o Llama-3.1-8B, sugerindo que ganhos de especialização dependem fortemente da qualidade do modelo base [Grattafiori et al. 2024].

Em termos de escala, observa-se uma tendência geral de melhoria de desempenho com o aumento do número de parâmetros. A principal exceção ocorre no DATASUS (SIH/SUS), onde o Qwen2.5-Coder-14B supera o modelo de 32B (EX: 27.5% vs. 23.5%), indicando que modelos menores podem apresentar melhor adaptação em cenários específicos e mais desafiadores.

Os resultados evidenciam também uma diferença consistente entre as métricas de Correspondência de Componentes (CM) e Acurácia de Execução (EX) em todos os modelos e bases. Enquanto os valores de CM frequentemente superam 55%, os valores de EX são substancialmente inferiores, especialmente em bases mais complexas como o DATASUS. Esse padrão indica que os modelos conseguem identificar corretamente compo-

nentes individuais da consulta, mas falham na sua composição em consultas executáveis, evidenciando limitações de **generalização composicional**, isto é, a capacidade de combinar corretamente componentes conhecidos em estruturas não vistas durante o treinamento [Gan et al. 2022].

Por fim, observamos um ***gap* expressivo entre bases tradicionais e cenários reais**. Enquanto os modelos alcançam até 70.7% de EX no Spider, o melhor resultado no DATASUS não ultrapassa 27.5%, representando uma queda relativa superior a 60%. Esse resultado indica que o desempenho medido em *benchmarks* tradicionais não se transfere diretamente para contextos reais.

Essa discrepância pode ser explicada por três fatores principais: (i) maior complexidade estrutural no DATASUS, com média de 15 tabelas por banco (vs. 5–6 nas demais bases), aumentando a dificuldade de junções; (ii) presença de vocabulário técnico de domínio, que dificulta o alinhamento semântico entre pergunta e esquema; e (iii) uso integral do português no esquema e nos dados, contrastando com o pré-treinamento predominantemente em inglês dos modelos. Em contraste, o Spider apresenta esquemas mais simples e menos representativos de cenários reais [Lei et al. 2025], o que pode levar a estimativas excessivamente otimistas de desempenho.

5.2. Análise Por Dificuldade

A partir da análise dos resultados da Tabela 3 sob a perspectiva da dificuldade das consultas, observa-se uma **degradação sistemática de desempenho** à medida que a complexidade aumenta. Esse comportamento é consistente em todas as bases, mas particularmente acentuado em cenários mais desafiadores. No Spider (dev), por exemplo, o melhor modelo (Qwen2.5-Coder-32B) apresenta queda de 86.5% em consultas fáceis para 39.2% nas extra-difíceis. No DATASUS (SIH/SUS), essa degradação é ainda mais severa: o melhor modelo (Qwen2.5-Coder-14B) cai de 64.7% para apenas 6.7%.

Embora essa tendência seja esperada — dado que o aumento de dificuldade implica maior número de operações SQL, junções e subconsultas —, os resultados indicam que a degradação é **desproporcional**, especialmente em bases reais. Esse comportamento reforça a evidência já observada na análise agregada: os modelos são capazes de identificar componentes individuais das consultas, mas falham em sua composição correta. Pequenas inconsistências estruturais, como erros em condições de junção ou filtros mal posicionados, são suficientes para invalidar a execução da consulta, evidenciando limitações de **generalização composicional**.

Ao analisar os resultados por faixa de dificuldade, o DATASUS (SIH/SUS) se destaca como o cenário mais desafiador em todos os níveis. Mesmo nas consultas fáceis, o melhor resultado de EX não ultrapassa 64.7%, enquanto nas demais bases valores superiores a 78% são alcançados. À medida que a dificuldade aumenta, essa diferença se amplia significativamente: nenhum modelo ultrapassa 10.5% de EX na dificuldade média, e praticamente não há acertos em consultas difíceis. Esses resultados indicam que a dificuldade das consultas no DATASUS não decorre apenas da estrutura SQL, mas também de fatores adicionais, como maior complexidade do esquema e vocabulário técnico de domínio, que tornam o alinhamento semântico mais desafiador.

A comparação entre modelos generalistas e especializados em código reforça esse diagnóstico. Embora modelos da família Qwen2.5-Coder apresentem desempenho

superior na maioria dos cenários, seus ganhos diminuem significativamente em consultas mais complexas e em bases mais desafiadoras. Isso sugere que a especialização em código melhora a geração estrutural, mas não resolve o principal gargalo da tarefa: o alinhamento entre linguagem natural, esquema e contexto de domínio.

Em conjunto, esses resultados mostram que os modelos atuais são mais eficazes em resolver aspectos locais da tarefa do que em realizar raciocínio estruturado completo. Esse padrão reforça a necessidade de avanços em **raciocínio composicional** e **robustez semântica**, além de destacar a importância de avaliações em múltiplos domínios para capturar de forma mais realista as limitações dos modelos.

5.3. Discussão e Principais *Insights*

A partir dos resultados, destacamos três implicações centrais para a tarefa Text-to-SQL em português:

1. **Avaliações em *benchmarks* tradicionais superestimam desempenho.** Observamos um *gap* substancial entre cenários tradicionais e reais: enquanto modelos alcançam mais de 65% de EX no Spider, o melhor resultado no DATASUS não ultrapassa 27.5%. Essa queda superior a 60% indica que avaliações baseadas em *benchmarks* isolados podem superestimar sistematicamente a capacidade dos modelos, reforçando a necessidade de *benchmarks* multi-domínio.
2. **O principal gargalo é a generalização composicional.** A diferença consistente entre CM e EX mostra que os modelos identificam corretamente componentes individuais das consultas, mas falham em sua composição em consultas executáveis. Isso indica que o desafio central não está na geração sintática de SQL, mas na capacidade de combinar corretamente múltiplos elementos em estruturas mais complexas.
3. **Especialização em código é insuficiente em cenários reais.** Modelos especializados, como Qwen2.5-Coder, apresentam ganhos consistentes em geração estrutural, mas esses ganhos diminuem em bases mais complexas. Isso sugere que o fator limitante passa a ser o alinhamento semântico entre linguagem natural e esquema, especialmente em domínios com vocabulário técnico.

Em conjunto, esses resultados mostram que cenários reais permanecem um desafio em aberto. Avanços futuros dependem não apenas de modelos mais capazes, mas de melhor adaptação a domínio e de mecanismos que favoreçam raciocínio estruturado. Essas descobertas indicam que o avanço da tarefa depende menos de ganhos incrementais em modelos e mais de abordagens que promovam alinhamento semântico robusto e raciocínio composicional em cenários reais.

6. Conclusão e Trabalhos Futuros

Este trabalho apresentou a primeira avaliação sistemática e unificada da tarefa de Text-to-SQL em português sob uma perspectiva multi-domínio, endereçando duas limitações centrais da literatura: a fragmentação das bases de dados disponíveis e a ausência de protocolos experimentais comparáveis. Ao consolidar diferentes cenários em um único *benchmark* e adotar um protocolo padronizado de avaliação, este estudo permite, pela primeira vez, uma análise controlada do desempenho de LLMs em contextos diversos.

Como primeira contribuição (QP1), construímos um *benchmark* unificado a partir de quatro conjuntos de avaliação — Spider-pt (*dev* e *test*), text2SQL4PM e DATASUS (SIH/SUS) — cobrindo diferentes domínios, níveis de complexidade e características estruturais. Essa consolidação evidencia que a tarefa de Text-to-SQL em português não é homogênea, mas fortemente dependente do domínio, da estrutura do esquema e do alinhamento linguístico.

Como segunda contribuição (QP2), realizamos uma avaliação sistemática de seis LLMs de código aberto sob um protocolo experimental controlado. Os resultados indicam a dominância dos modelos da família Qwen2.5, com destaque para o Qwen2.5-Coder-32B, e confirmam que a especialização em código contribui para a geração estrutural de consultas. No entanto, mostramos que esses ganhos são limitados em cenários mais complexos, evidenciando que avanços arquiteturais ou de escala não são suficientes para resolver o problema em sua totalidade.

A principal descoberta deste trabalho é a identificação de um ***gap* substancial e sistemático entre *benchmarks* tradicionais e cenários reais**. Enquanto modelos alcançam mais de 65% de EX no Spider, o desempenho no DATASUS não ultrapassa 27.5%, representando uma queda relativa superior a 60%. Esse resultado demonstra que avaliações baseadas em *benchmarks* isolados podem superestimar significativamente a capacidade dos modelos, especialmente em contextos reais caracterizados por maior complexidade estrutural e vocabulário de domínio.

Além disso, nossos resultados mostram que os modelos atuais são significativamente mais eficazes em identificar componentes isolados das consultas do que em compô-los corretamente em estruturas executáveis. Essa discrepância entre CM e EX evidencia que o principal gargalo da tarefa não está na geração sintática de SQL, mas na **generalização composicional** e no alinhamento semântico entre linguagem natural e esquemas complexos. Em outras palavras, o desafio central da tarefa é menos sobre "gerar SQL" e mais sobre "compreender e estruturar corretamente a informação".

Em conjunto, esses achados indicam que o avanço da tarefa de Text-to-SQL em português depende não apenas de modelos mais capazes, mas de avaliações mais realistas e de abordagens que promovam melhor adaptação a domínio e raciocínio estruturado. Nesse sentido, este trabalho reforça a importância de *benchmarks* multi-domínio como ferramenta essencial para medir progresso de forma confiável.

Este estudo apresenta algumas limitações. A avaliação foi conduzida sob uma única configuração de *prompting (zero-shot)* e decodificação determinística, não explorando estratégias como *few-shot* ou *chain-of-thought*, que podem impactar o desempenho dos modelos. Além disso, não incluímos modelos proprietários de última geração, como GPT-4 ou Gemini, que têm apresentado resultados superiores em trabalhos recentes. Por fim, o tamanho reduzido de algumas bases, como o DATASUS, pode introduzir maior variabilidade nas estimativas de desempenho. Ainda assim, acreditamos que o protocolo adotado oferece uma base sólida, reprodutível e comparável para análises sistemáticas.

Como trabalhos futuros, pretendemos expandir o *benchmark* com novas bases em português, incluindo domínios como educação e finanças, além de avaliar modelos proprietários e investigar o impacto de estratégias mais avançadas de *prompting* e raciocínio.

Além disso, planejamos conduzir avaliações qualitativas para um entendimento aprofundado dos erros dos modelos. Mais amplamente, esperamos que este trabalho contribua para o desenvolvimento de avaliações mais rigorosas e para uma compreensão mais realista das limitações atuais dos sistemas *Text-to-SQL* em cenários do mundo real.

Uso de Inteligência Artificial

Este trabalho utilizou os modelos Claude, ChatGPT e Gemini para revisão gramatical e tradução do resumo para o inglês, com supervisão e revisão dos autores.

Agradecimentos

Este trabalho foi parcialmente financiado pelo Instituto Nacional de Ciência e Tecnologia em Inteligência Artificial Responsável para Linguística Computacional, Tratamento e Disseminação de Informação – INCT TILDIAR (#408490/2024-1),o presente trabalho foi realizado com apoio da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior – Brasil (CAPES) – Código de Financiamento 001, a Fundação de Amparo à Pesquisa do Estado de Minas Gerais (FAPEMIG), Dadosfera, Embrapii e SEBRAE. Os autores também agradecem à NVIDIA pela disponibilização da infraestrutura computacional utilizada na execução de parte dos experimentos apresentados neste trabalho.

Referências

Centenaro, B. R. A. (2025). IA generativa para consultas SQL a partir de linguagem natural: Uma avaliação utilizando dados educacionais brasileiros. *Trabalho de Conclusão de Curso (Bacharelado), Universidade Federal de Santa Catarina https://repositorio.ufsc.br/handle/123456789/266444*.

de Carvalho, L. F. C., Júnior, P. S. d. S., and de Oliveira, H. T. A. (2025). Benchmarking large language models for text-to-sql in brazilian portuguese and english. In *Simpósio Brasileiro de Tecnologia da Informação e da Linguagem Humana (STIL)*, pages 101–112.

Dettmers, T., Pagnoni, A., Holtzman, A., and Zettlemoyer, L. (2023). Qlora: Efficient finetuning of quantized llms. *Advances in neural information processing systems*, 36:10088–10115.

Dou, L., Gao, Y., Pan, M., Wang, D., Che, W., Zhan, D., and Lou, J.-G. (2023). Multispider: towards benchmarking multilingual text-to-sql semantic parsing. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 12745–12753.

Fróes, K. and Braghetto, K. (2025). Exploring temporal text-to-sql challenges in brazilian portuguese: Lessons from educational data. In *Proceedings of the Brazilian Symposium on Data Bases (SBBD)*, pages 963–969.

Gan, Y., Chen, X., Huang, Q., and Purver, M. (2022). Measuring and improving compositional generalization in text-to-SQL via component alignment. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 831–843. Association for Computational Linguistics (ACL).

Gao, D., Wang, H., Li, Y., Sun, X., Qian, Y., Ding, B., and Zhou, J. (2024). Text-to-sql empowered by large language models: A benchmark evaluation. *Proc. VLDB Endow.*, 17(5):1132–1145.

Grattafiori, A., Dubey, A., Jauhri, A., Pandey, A., Kadian, A., Al-Dahle, A., Letman, A., Mathur, A., Schelten, A., Vaughan, A., et al. (2024). The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Hui, B., Yang, J., Cui, Z., Yang, J., Liu, D., Zhang, L., Liu, T., Zhang, J., Yu, B., Lu, K., et al. (2024). Qwen2. 5-coder technical report. *arXiv preprint arXiv:2409.12186*.

José, M. A. and Cozman, F. G. (2021). mrat-sql+ gap: a portuguese text-to-sql transformer. In *Brazilian Conference on Intelligent Systems*, pages 511–525. Springer.

Jose, M. A. and Cozman, F. G. (2023). A multilingual translator to sql with database schema pruning to improve self-attention. *International Journal of Information Technology*, 15(6):3015–3023.

Lei, F., Chen, J., Ye, Y., Cao, R., Shin, D., Su, H., Suo, Z., Gao, H., Hu, W., Yin, P., et al. (2025). Spider 2.0: Evaluating language models on real-world enterprise text-to-sql workflows. In *International Conference on Learning Representations (ICLR)*, pages 28691–28735.

Li, J., Hui, B., Qu, G., Yang, J., Li, B., Li, B., Wang, B., Qin, B., Geng, R., Huo, N., et al. (2023). Can llm already serve as a database interface? a big bench for large-scale database grounded text-to-sqls. *Advances in Neural Information Processing Systems*, 36:42330–42357.

Moraes, M., Figueiredo, I., Marques, V., Santos, J., and Manssour, I. H. (2025). From questions to answers: A natural language interface for datasus hospitalization data. In *Escola Regional de Aprendizado de Máquina e Inteligência Artificial da Região Sul (ERAMIA-RS)*, pages 296–299. SBC.

Pedroso, B. C., Pereira, M. R., and Pereira, D. A. (2025). Performance evaluation of llms in the text-to-sql task in portuguese. In *Simpósio Brasileiro de Sistemas de Informação (SBSI)*, pages 260–269.

Petrola, L. and Franco, W. (2025). Heuristic-guided text-to-sql translation with llms: Optimizing natural language interfaces for relational databases. In *Proceedings of the Brazilian Symposium on Data Bases (SBBD)*, pages 128–141.

Pham, K. T., Nguyen, T. H., Jo, J., Nguyen, Q. V. H., and Nguyen, T. T. (2025). Multilingual text-to-sql: Benchmarking the limits of language models with collaborative language agents. In *Australasian Database Conference*, pages 108–123. Springer.

Qwen Team (2025). Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*.

Song, Y., Wang, G., Li, S., and Lin, B. Y. (2025). The good, the bad, and the greedy: Evaluation of llms should not ignore non-determinism. In *Proceedings of the Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 4195–4206.

Yamate, B. Y., Neubauer, T. R., Fantinato, M., and Peres, S. M. (2025). Text-to-sql oriented to the process mining domain: A pt-en dataset for query translation. *arXiv preprint arXiv:2509.09684*.

Yu, T., Zhang, R., Yang, K., Yasunaga, M., Wang, D., Li, Z., Ma, J., Li, I., Yao, Q., Roman, S., et al. (2018). Spider: A large-scale human-labeled dataset for complex and cross-domain semantic parsing and text-to-sql task. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 3911–3921.