

CLEAR-SQL: 面向 Text-to-SQL 语义解析的 跨语言增强自适应推理框架

章概煜, 刘梦赤, 尹 康

(华南师范大学 计算机学院, 广东 广州 510631)

摘要: 为解决 Text-to-SQL 在多语言应用中面临的零样本性能受限、语义与模式失配等挑战, 提出一种跨语言增强自适应推理框架 CLEAR-SQL。该框架集成了 3 个核心组件: ① 混合对比检索, 通过多语言句向量模型统一 SQL 结构模式与跨语言语义对齐; ② 分层模式推理链, 利用多轮思维链进行表、列及连接关系的推导; ③ 执行反馈循环, 结合类型化的错误校正机制进行查询验证与迭代优化。实验结果表明, 该框架在 CSpider 基准测试中, 精确匹配率 (EM) 达到 72.1%; 在 Multispider 基准的 6 种语言上, 平均 EM 达到 72.3%, 分别超越强基线 10.3 和 11.7 个百分点。此外, 混合检索将 SQL 骨架相似度提升了 22.3 个百分点, 错误校正对 SQL 语法错误的校正率达到 65.6%。研究结论证实, 所提出的框架能够有效提升多语言 Text-to-SQL 解析的准确性和鲁棒性, 在无需并行资源的情况下实现了语言无关的解析能力, 为构建多语言数据库自然语言接口提供了新的技术标准。

关键词: Text-to-SQL; 跨语言语义解析; 大语言模型; 混合对比检索; 多轮思维链; 执行反馈循环

DOI: 10.11907/rjdk.251541

扫描二维码阅读全文: 

中图分类号: TP391.1

文献标识码: A

文章编号: 1672-7800(XXXX)0XX-0001-09

CLEAR-SQL: Cross-Language Enhanced Adaptive Reasoner Framework for Text-to-SQL Semantic Parsing

ZHANG Gaiyu, LIU Mengchi, YIN Kang

(School of Computer Science, South China Normal University, Guangzhou 510631, China)

Abstract: To address the challenges in multilingual Text-to-SQL applications, such as limited zero-shot performance and semantic-schema misalignment, a cross-language enhanced adaptive reasoner framework, CLEAR-SQL, is proposed. The framework integrates three core components: ① Hybrid contrastive retrieval, which unifies SQL structural patterns with cross-lingual semantic alignment by leveraging multilingual sentence embedding models; ② A hierarchical schema reasoning chain that utilizes a multi-turn Chain-of-Thought for table, column, and join deduction; and ③ An execution feedback loop that incorporates a typed error correction mechanism for query validation and iterative optimization. Experimental results demonstrate that the framework achieves an Exact Match (EM) rate of 72.1% on the CSpider benchmark and an average EM of 72.3% across six languages on the Multispider benchmark, surpassing strong baselines by 10.3 and 11.7 percentage points, respectively. Furthermore, the hybrid retrieval enhances SQL skeleton similarity by 22.3 percentage points, and the error correction mechanism attains a 65.6% correction rate for SQL syntax errors. The study concludes that the proposed framework effectively improves the accuracy and robustness of multilingual Text-to-SQL parsing, establishing language-agnostic parsing capabilities without parallel resources and setting a new technical standard for building multilingual database natural language interfaces.

Key Words: Text-to-SQL; cross-lingual semantic parsing; large language models; hybrid contrastive retrieval; multi-turn Chain-of-Thought (CoT); execution feedback loop

收稿日期: 2025-09-01

基金项目: 国家自然科学基金项目 (61672389); 广州市大数据智能教育重点实验室项目 (201905010009)

作者简介: 章概煜 (2001-), 男, CCF 学生会员, 华南师范大学计算机学院硕士研究生, 研究方向为跨语言 Text-to-SQL 语义解析; 刘梦赤 (1962-), 男, 华南师范大学计算机学院教授, 研究方向为数据库语义推理、大数据系统; 尹康 (2001-), 男, 华南师范大学计算机学院硕士研究生, 研究方向为基于深度学习的时间序列预测方法。本文通讯作者: 章概煜。

0 引言

在当今大数据时代,数据已成为推动各行各业发展的核心要素^[1]。尽管关系型数据库广泛应用于数据存储与管理,但普通用户在使用SQL等数据库查询语言时常常面临较高的技术门槛。编写复杂的SQL查询需要专业的知识和经验。因此,如何有效地将自然语言转换为SQL查询,成为自然语言处理(Natural Language Processing, NLP)领域的重要研究课题^[2]。

Text-to-SQL技术旨在通过将自然语言查询自动转换为SQL命令,极大地降低数据访问的复杂性^[3]。尽管像ChatGPT-4^[4]这样的大语言模型(Large Language Models, LLM)在英文基准(如Spider^[5])上取得了显著成就,但这种通用能力在直接应用于复杂、专业且高度结构化的数据库查询任务时,仍然面临着关键瓶颈。例如:在面对缺少良好注释、包含大量表和列的真实数据库时,难以进行精确的模式链接;在生成需要复杂嵌套、多重连接或特定领域函数的SQL时,准确率显著下降;其输出具有随机性且缺乏有效的验证与修正机制。因此,当前研究的焦点已从简单地调用LLM,转向如何构建一个能够引导、约束并增强LLM在Text-to-SQL这一专业领域表现的专用框架。

将这些挑战从英语环境迁移至多语言场景时,上述问题变得尤为突出^[6-7]。其中,最核心的挑战源于语义与模式的失配。语言的结构差异和翻译歧义,常常破坏了自然语言表达与数据库模式间的精确对齐,直接导致模型在低资源语言上零样本性能受限,并限制了上下文学习(In-Context Learning, ICL)的跨语言推理效能。此外,多语言基准的缺乏也阻碍了该领域的系统性评估与发展^[8]。

1 相关工作

近年来,大型语言模型(Large Language Model, LLM)的兴起极大地推动了零样本跨语言语义解析的发展,尤其是在Text-to-SQL这一复杂任务上。研究工作主要沿着两条技术路径展开:①利用LLM进行高质量的数据增强;②通过检索增强的ICL直接进行跨语言推理^[9]。

1.1 基于LLM的数据增强范式

在缺乏目标语言标注数据的情况下,一个主流策略是通过“翻译—训练”范式生成合成数据。传统方法依赖于通用机器翻译系统,但往往因领域不匹配而效果不佳。近期,研究者开始利用LLM强大的少样本学习能力以革新这一范式。

Awasthi等^[10]通过少量种子示例进行少样本提示,可以引导LLM高效地将英文数据集翻译至目标语言,并通过采样和过滤提升数据质量。进一步地,Do等^[11]引入了Zero-MParser框架,引导LLM增量式地构建目标语言的逻辑

形式,从而生成结构更准确的合成数据。然而,这些方法共同的局限在于,仍需离线生成静态数据集,不仅计算成本高,也未能充分利用LLM的实时推理能力。

1.2 检索增强的跨语言上下文学习

为摆脱对静态数据集的依赖,检索增强的ICL范式成为更前沿的解决方案^[12-15]。该范式在推理时为非英语查询动态检索英文示例以构建提示。Shi等^[16]提出的XRICL框架是这一方向的代表性工作,它通过“检索—重排序”流水线从英文候选池中选择最相关的示例。进一步地,Tanwar等^[17]的研究强调,通过任务对齐器进行显式的标签空间映射对提升跨语言ICL性能至关重要,这揭示了结构化对齐的必要性。

1.3 本文方法创新

综上所述,现有研究在数据增强和检索增强两个方向上均取得了显著进展。然而,数据增强范式存在成本高、无法实时推理的局限;而现有的检索增强方法在结构对齐精确性、深度模式推理能力和生成结果鲁棒性方面仍有待提升。

为系统性地解决这些问题,本文提出了一个名为CLEAR-SQL的跨语言增强自适应推理框架。与现有方法相比,该框架的核心创新与改进之处在于:①提出了一种更普适的混合对比检索机制,它在无需并行语料的情况下,首次将语义相似性与SQL结构相似性相结合,以检索更具参考价值的示例;②设计了分层模式推理链,通过多轮CoT引导LLM进行结构化的模式推理(表→列→连接)^[18-19],以应对复杂数据库模式下的语义失配问题,这是现有检索增强方法普遍缺乏的深度推理能力;③引入了执行反馈环路,通过验证和迭代修正LLM的生成错误,显著提升了最终SQL查询的可用性和准确性,解决了现有ICL方法结果不可靠的痛点。

综上,CLEAR-SQL通过一个更完整、更鲁棒的直接推理框架,为零样本跨语言Text-to-SQL解析提供了新的解决方案。

2 模型构建

为系统性地应对跨语言Text-to-SQL中的各项挑战,本文提出了CLEAR-SQL模型,其核心由4个紧密协作的模块构成:语义对齐与候选SQL生成、分层模式推理链、混合对比检索,以及执行反馈与迭代修正。本文方法具体流程如图1所示。

2.1 语义对齐与候选SQL生成

该模块旨在为输入的非英语查询,从高资源的英文示例池中高效地初步筛选出一组语义上高度相关的候选示例。其核心目标是有效地将输入问题的语义意图与训练示例的语义意图对齐。尽管源语言和目标语言在词汇和语法上存在差异,但其底层表达的语义等价性使得LLM能

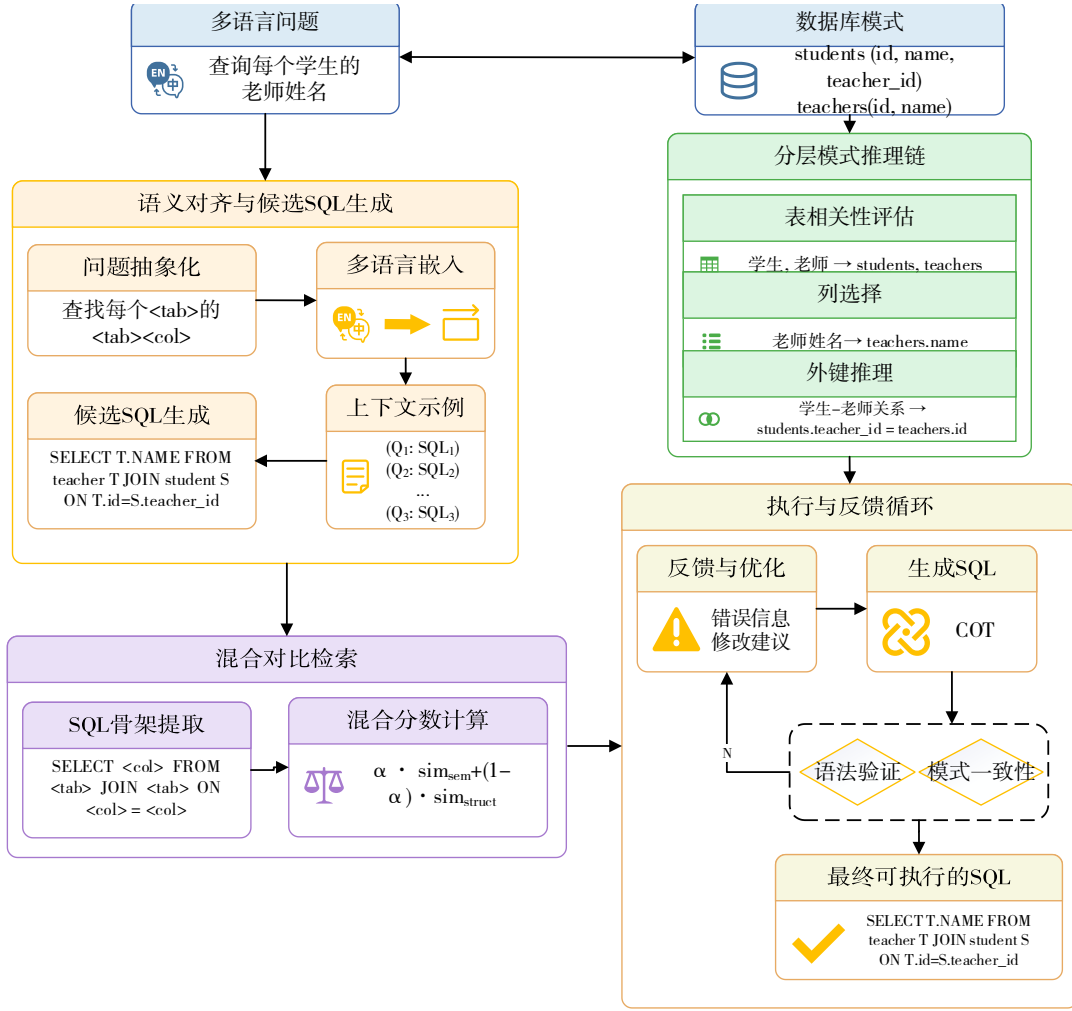


Fig. 1 The overall architecture of CLEAR-SQL model

图1 CLEAR-SQL模型整体架构

够捕捉其共享的内在含义。这一能力根植于Chang等^[20]所揭示的共享高维语义空间,它允许LLM有效地进行跨语言的知識迁移。

为充分利用此特性以增强ICL,本文提出通过选择语义上高度相似的示例以构建提示。源查询与目标示例间的语义相似度越高,该示例作为上下文提示的有效性就越强。实现这一目标的关键在于消除领域特定信息的干扰,并构建一个统一的跨语言语义空间。

为剥离数据库模式中特定于领域的词汇(如表名、列名),对输入查询 Q 进行抽象化处理。不同于传统的基于固定规则的掩码方法,本文利用大型语言模型的上下文学习能力以实现动态、上下文感知的抽象化处理。通过在提示中提供少样本掩码的示例,让LLM自主学习并推断出掩码策略,将查询中与数据库模式 $D=(T, C)$ 相关的实体(表、列)及字面量值 $\mathcal{V}_{\text{literal}}$,分别替换为<tab>、<col>和<val>等通用占位符。该策略在概念上可由如式(1)所示的替换函数 $MASK(x)$ 表示。

$$MASK(x) = \begin{cases} < \text{tab} > & \text{if } x \in T \\ < \text{col} > & \text{if } x \in C \\ < \text{val} > & \text{if } x \in \mathcal{V}_{\text{literal}} \\ x & \text{otherwise} \end{cases} \quad (1)$$

通过此函数,原始查询被转换为抽象化查询,此过程旨在保留查询的结构与意图,同时消除词汇层面的异构性。

在获得抽象化查询后,采用多语言Sentence-BERT将其嵌入到共享的跨语言语义空间中^[21]。给定查询 q 及其候选示例 q_i ,其嵌入向量分别为 v_q 和 v_{q_i} 。两者之间的相似度 $\text{sim}_{\text{sem}}(q, q_i)$ 通过归一化的欧氏距离进行度量,如式(2)所示。

$$\text{sim}_{\text{sem}}(q, q_i) = 1 - \frac{\|v_q - v_{q_i}\|_2}{\max(\|v_q\|_2, \|v_{q_i}\|_2)} \quad (2)$$

其中, $\|\cdot\|_2$ 表示L2范数运算。这种归一化的欧几里德距离度量确保了不同语言之间的尺度不变相似性度量。基于此相似度得分,系统检索出top- k 个最相关的候选示例,并利用这些示例生成初步的候选SQL查询,为后续精细化处理提供输入。

2.2 分层模式推理链

为解决跨语言场景下普遍存在的语义与模式失配问题,本文设计了分层模式推导模块。该模块通过一系列精心设计的 CoT 提示,引导 LLM 对数据库模式进行结构化、由粗到细的推理,最终生成一个精简且与查询高度相关的“推理后模式” $\mathcal{D}_{\text{reasoned}}$ 。该过程模拟了数据库专家的分析逻辑,包括以下 3 个核心步骤:

(1)表相关性评估。LLM 通过以下方式识别与输入问题语义一致的表:首先通过 LLM 的跨语言语义理解进行语义匹配(例如,“部门”→department,“秘书”→head);其次,通过 LLM 的跨语言隐式关系检测进行上下文推理(例如,“管理的”→management)。

(2)列选择。对于每个相关表,LLM 执行双重识别:首先通过查询意图分析查询目标识别(例如,“创建年份”→department.creation for);其次,通过 WHERE/JOIN 条件分析进行约束模式识别(例如,“出生于河南”→head.born_state)。

(3)外键推理。根据表相关性评估和列选择的结果,推断表间关系,以确保有效的 JOIN 操作。

2.3 混合对比检索

在初步语义筛选和模式推理的基础上,该模块旨在通过更精细的度量,对上下文示例进行二次优化,以确保最终生成的 SQL 在语法结构上与用户意图高度一致。为此,本文提出了一种创新的混合对比检索机制。

为量化 SQL 查询间的结构相似性,为每个 SQL 查询提取其“结构骨架” $\mathcal{P}$ 。该骨架通过确定性替换函数 $\text{Replace}(x)$,将 SQL 语句中的模式元素、SQL 操作符和字面量值替换为统一的占位符。 $\text{Replace}(x)$ 的定义如式(3)所示。

$$\text{Replace}(x) = \begin{cases} \langle \text{tab} \rangle & \text{if } x \in \mathcal{T} \\ \langle \text{col} \rangle & \text{if } x \in \mathcal{C} \\ \langle \text{op} \rangle & \text{if } x \in \mathcal{V}_{\text{op}} \\ \langle \text{val} \rangle & \text{if } x \in \mathcal{V}_{\text{literal}} \\ x & \text{otherwise} \end{cases} \quad (3)$$

其中, $\mathcal{T}$ 代表数据库表名, $\mathcal{C}$ 表示数据库列名。 $\mathcal{V}_{\text{op}}$ 表示 SQL 运算符, $\mathcal{V}_{\text{literal}}$ 表示带引号的字符串和数字文字。通过此过程,将一个具体的 SQL 查询转换成了一个抽象的结构化表示。

随后,两个 SQL 查询的结构相似度 $\text{sim}_{\text{struct}}$,通过其骨架集合 $\mathcal{P}_i$ 和 $\mathcal{P}_j$ 的杰卡德相似系数(Jaccard Similarity)进行计算,如式(4)所示。

$$\text{sim}_{\text{struct}} = \frac{|\mathcal{P}_i \cap \mathcal{P}_j|}{|\mathcal{P}_i \cup \mathcal{P}_j|} \quad (4)$$

最终,通过引入一个可学习的权重超参数 α ,将语义相似度 sim_{sem} 与结构相似度 $\text{sim}_{\text{struct}}$ 进行线性组合,得到混合相似度得分 $\text{sim}_{\text{hybrid}}$,如式(5)所示。

$$\text{sim}_{\text{hybrid}} = \alpha \cdot \text{sim}_{\text{sem}} + (1 - \alpha) \cdot \text{sim}_{\text{struct}} \quad (5)$$

该混合得分能够综合评估候选示例在语义和结构两个维度上的匹配度。系统据此对候选示例进行最终的重

排序,筛选出最高质量的示例集 E_k ,为后续 SQL 生成提供最优上下文。

2.4 执行反馈与迭代修正

为保障框架输出的 SQL 查询的最终可用性与正确性,本文设计了执行反馈与迭代修正环路,该模块是系统的最后一道质量防线。

2.4.1 上下文感知 SQL 生成

系统整合前序模块产出的高质量示例集 E_k 和推理后模式 $\mathcal{D}_{\text{reasoned}}$ 构建最终提示,并提交给 LLM 以生成目标问题 Q 的查询 SQL_{final} ,如式(6)所示。

$$SQL_{\text{final}} = \text{LLM}(Q, E_k, \mathcal{D}_{\text{reasoned}}) \quad (6)$$

2.4.2 双重验证与自我修正机制

生成的 SQL_{final} 将经过严格的双重验证:

(1)语法验证。利用 SQL 解析器将生成的查询构建为抽象语法树(AST)。如果 AST 无法成功构造(如出现 SELECT 后缺少 FROM 等结构性错误),则判定为语法无效,并捕获解析器返回的精确错误信息用于后续修正。

(2)模式一致性验证。在 AST 成功构造后,遍历 AST 的节点,系统性地提取出所有被引用的表名和列名。随后,将这些提取出的实体与已知的数据库模式 $\mathcal{D}$ 进行比对。若发现任何在模式 $\mathcal{D}$ 中不存在的表或列(如查询了 teachers.salary,而该列不存在),则判定为模式不一致。

如果在验证过程中检测到错误,模型将构造错误查询 SQL_{err} 、错误消息 $\mathcal{E}_{\text{msg}}$ 、修正建议 $\text{adv}_{\text{correct}}$,以及完整的数据库模式 $\mathcal{D}$ 反馈给 LLM,引导其进行自我修正,如式(7)所示。

$$SQL_{\text{new}} = \text{LLM}(SQL_{\text{err}}, \mathcal{E}_{\text{msg}}, \text{adv}_{\text{correct}}, \mathcal{D}) \quad (7)$$

为更直观地阐释式(7)的运行机制,给出一个具体示例。假设模型针对问题“查询所有教师的薪水”生成了错误的 SQL 查询“SELECT salary FROM teachers (SQL_{err})”,验证阶段会因 teachers 表中不存在 salary 列而失败,并捕获错误信息“Table-column mismatch($\mathcal{E}_{\text{msg}}$)”。系统据此生成修正建议“There is not column-salary in table-teachers, please consider tables-join ($\text{adv}_{\text{correct}}$)”,LLM 在接收到这些完整的上下文信息后,便能生成修正后的正确查询 SELECT T2.salary FROM teachers AS T1 JOIN salaries AS T2 ON T1.id=T2.teacher_id(SQL_{new})。此迭代过程将持续进行,直至生成的 SQL 通过验证,或达到预设的最大尝试次数。该闭环机制显著提升了框架鲁棒性。

3 实验与结果分析

3.1 数据集

本文在 3 个广泛使用的公开基准数据集上进行实验,这些数据集覆盖了从单语到多语、从简单到复杂的多种 Text-to-SQL 场景。

(1)Spider。一个大规模、跨领域的英文 Text-to-SQL

基准数据集。它包含横跨 200 个不同数据库的 10 181 个问题, 及其复杂的查询逻辑, 如多表连接、嵌套查询、复杂聚合等。遵循其官方的数据划分, 将 Spider 的训练集作为本文模型进行示例检索的英文候选池。

(2) CSpider^[22]。它是 Spider 的中文翻译版本, 增加了额外的语言适应性, 同时保持了相同的模式结构和 SQL 的复杂性。CSpider 的中文翻译保留了英文模式结构, 并带来了包括问题到数据库映射和分词歧义在内的跨语言挑战。

(3) Multispider^[23]。它是一个更全面的多语言 Text-to-SQL 基准, 其在 Spider 和 CSpider 的基础上, 进一步扩展到了日语、法语、西班牙语、德语等多种语言, 这使得对跨语言迁移能力的系统评估成为可能。

3.2 实验设置

本文 CLEAR-SQL 框架主要基于 LLM 实现。在实验中, 采用了 DeepSeek-R1 (671 B 参数、64 K 上下文) 模型作为核心的推理引擎^[24]。对于语义相似度计算, 采用了社区广泛使用的 paraphrase-multilingual-MiniLM-L12-v2 模型, 该模型支持超过 50 种语言, 能够高效地进行跨语言句向量表征。

3.2.1 实验环境

本文核心框架 CLEAR-SQL 采用基于上下文学习 (Learning, ICL) 的技术路径, 主要依赖于对 LLM API 的调用, 实验过程不涉及本地模型训练或微调。因此, 所有实验均在一台标准的 CPU 服务器上完成, 具体配置为 Intel (R) Xeon (R) Platinum 8260 CPU @ 2.40 GHz, 配备 512 GB 系统内存, 操作系统为 CentOS 7.9。这种轻量级的部署要求使得该框架具有极高的可用性和经济性。

作为对比, 参考部分先进基线模型如 XiYan-SQL, 其采用了基于模型微调的技术路径。根据其官方实现, 复现该类模型通常需要较高的计算资源, 复现方案在如两台 NVIDIA A6000 上进行两天的模型训练。通过明确本文框架与此类方法的资源差异, 旨在更全面地评估 CLEAR-SQL 在实现具有竞争力的性能的同时, 在计算效率和部署成本上所体现出的显著优势。

对所有测试的实验设置进行了标准化。最大 Token 限制设置为 2 048。对于 CLEAR-SQL 管道的所有 SQL 生成任务, 使用温度 0.7 和 $k=9$ 的 top- k 示例采样。通过对验证集进行网格搜索, 混合对比检索方程中 $\alpha=0.5$ 。

3.2.2 评价指标

使用官方评估脚本的精确匹配率 (Exact Match, EM) 作为核心性能度量, 要求生成的 SQL 在语法和语义上与标准答案完全等价, EM 是评估 Text-to-SQL 任务最严格和最常用的指标之一。

3.3 基线模型

为验证本文 CLEAR-SQL 框架的有效性与先进性, 选取了代表当前 Text-to-SQL 领域高水平的两种核心方法作

为基线, 进行直接性能对比。

(1) DAIL-SQL^[25]。它是选取的核心对比基线, 是一个在英文单语 Text-to-SQL 任务上取得领先性能的框架。其核心在于通过精细的提示工程, 特别是结合问题和 SQL 相似度的策略以选择上下文示例。通过与它的对比, 旨在评估一个高度优化的单语 SOTA 方法在直接应用于跨语言场景时的表现, 并以此凸显本文专为跨语言挑战设计模块的必要性与有效性。

(2) 领域 SOTA 模型。为确保评估的全面性, 还与代表不同技术范式的顶尖模型进行比较。在单语场景中, 对比 Graphix-3B + PICARD^[26]、SHIP + PICARD^[27] 和 FastRAT + AST Ranking^[28] 等基于微调的模型。在多语言场景中, 则对比基于“翻译—训练”范式的 RAT-SQL + XLM-R^[29-30]、mBART^[31] 等微调模型。通过与这些方法的比较, 可以将 CLEAR-SQL 置于更广阔的学术背景中, 衡量其相对于传统微调方法的竞争力。

此外, 为确保本文框架的时效性与前沿性, 将 CLEAR-SQL 的性能与近期在 BIRD 等复杂基准上取得 SOTA 性能的最新方法进行比较, 如 CHASE-SQL^[32] 和 XiYan-SQL^[33]。这些方法代表了当前领域在多候选生成、强化学习推理和多智能体协作等方向的最高水平。由于这些模型的复杂性和对特定资源的依赖性, 直接引用其在原论文中报告的最佳性能数据, 以进行公平、客观的横向对比, 从而更精准地定位本文框架的性能水平与贡献。

3.4 对比实验

3.4.1 在 Spider 上的结果

在英文标准基准 Spider 上评估 CLEAR-SQL 的性能, 以验证其基础能力, 实验结果如表 1 所示。

Table 1 Comparison of the exact matching rate with the current state-of-the-art method on the Spider dataset

表 1 在 Spider 数据集上与当前最先进方法的精确匹配率比较 (%)

模型	开发集	测试集
CLEAR-SQL	79.2	75.7
XiYan-SQL	76.2	74.1
Graphix-3B + PICARD	77.1	74.0
CHASE-SQL	75.7	73.7
DAIL-SQL	75.1	73.2
SHIP + PICARD*	77.2	73.1

如表 1 所示, CLEAR-SQL 在单语环境下同样具备 SOTA 竞争力, CLEAR-SQL 在开发集上实现了 79.2% 的 EM, 在测试集上实现了 75.7% 的 EM, 超越了包括核心基线 DAIL-SQL 在内的多种已发表的先进方法, 在已发表的方法中排名第二, 并与 CHASE-SQL (73.7%) 和 XiYan-SQL (74.1%) 等最新的 SOTA 框架相比同样表现出强劲的竞争力。这表明, 尽管框架的设计侧重于解决跨语言问题, 但其底层的检索与推理机制具有普适性, 在处理复杂的单语任务时同样表现出色。

3.4.2 在 CSpider 上的结果

跨语言性能是衡量本文框架核心价值的关键。在 CS-

pider 基准上对 CLEAR-SQL 进行评估,实验结果如表 2 所示。作为一个与 Spider 共享数据库模式的中文数据集,CSpider 能够直接反映模型在处理中英文语言对时,弥合语义鸿沟和解决模式失配问题的能力。

Table 2 Comparison of the exact matching rate with the current state-of-the-art method on the CSpider dataset

表 2 在 CSpider 数据集上与当前最先进方法的精确匹配率比较(%)

模型	开发集	测试集
CLEAR-SQL	73.4	70.1
XiYan-SQL	70.1	67.7
CHASE-SQL	69.7	65.9
FastRAT + AST Ranking	66.2	62.1
DAIL-SQL	65.2	61.3

如表 2 所示,CLEAR-SQL 在中文跨语言任务上取得了突破性进展,大幅领先于各类基线方法,CLEAR-SQL 在 CSpider 测试集上取得了 70.1% 的 EM,相较于核心基线 DAIL-SQL(61.3%)提升了惊人的 8.8 个百分点,并显著优于 FastRAT + AST Ranking 等其他先进方法,创造了该数据集的新 SOTA。更重要的是,与同样在跨语言任务上表现出色的 XiYan-SQL(67.7%)和 CHASE-SQL(65.9%)相比,本文框架依然保持了显著的领先地位。这一结果充分证明了为跨语言场景专门设计的模块,如语义抽象化和混合对比检索推导,能够有效解决由翻译和语言结构差异引起的语义与模式失配问题。

3.4.3 在 MultiSpider 上的结果

为了评估 CLEAR-SQL 在更广泛语言环境下的泛化能力和一致性,在 Multispider 上进行了实验,并将结果与主流方法及核心基线 DAIL-SQL 进行对比,实验结果如表 3 所示。

Table 3 Comparison of cross-lingual exact matching on the Multispider dataset(7 languages)

表 3 在 Multispider 上的跨语言精确匹配率比较(7 种语言)(%)

模型	英语	德语	西班牙语	法语	日语	中文	越南语	平均值(6 种语言)
单语训练 (Monolingual Training)								
mBART + SAVE	58.3	42.6	42.6	51.2	46.9	56.6	43.1	45.5
RAT-SQL + XLM-R + SAVE	68.8	63.9	62.7	65.7	54.3	66.2	66.1	63.2
DAIL-SQL	75.1	65.3	64.2	66.5	58.3	65.2	62.2	63.6
CLEAR-SQL	79.2	74.8	74.4	74.2	64.4	73.4	74.1	72.3
零样本直接预测 (Zero-Shot Directly Predict)								
mBART	-	50.9	52.2	50.7	43.1	49.6	45.3	48.6
XLM-R	-	57.6	60.8	59.1	48.3	55.5	56.5	56.3
DAIL-SQL	75.1	63.1	62.3	66.1	57.6	64.7	61.2	62.5
CLEAR-SQL	79.2	74.7	75.2	74.3	64.8	73.4	73.4	72.3

如表 3 所示,CLEAR-SQL 展现出强大的多语言泛化能力与语言无关性,性能远超各类基线。在涵盖 6 种非英语语言的 Multispider 基准上,CLEAR-SQL 的平均 EM 达到了 72.3%。这一性能不仅显著优于 RAT-SQL + XLM-R 等传统的微调方法(63.2%),也远超核心基线 DAIL-SQL 的推断性能(约 62.5%)。并且,DAIL-SQL 作为一个未针对跨

语言进行优化的单语框架,其性能在非英语语言上出现显著衰减。

更重要的是,CLEAR-SQL 在零样本直接预测和单语训练两种设置下性能几乎持平(均为 72.3%)。这有力地证明了其语言无关的设计优势,即框架无需针对特定目标语言进行训练,便能保持一致的高性能水平,具有极强的跨语言迁移能力。这一特性使得 CLEAR-SQL 在处理新的低资源语言时具有巨大的应用潜力。

3.4 消融实验

为定量地评估 CLEAR-SQL 框架中每个核心模块的独立贡献,在 CSpider 开发集上进行了一系列消融实验。以完整的 CLEAR-SQL 框架为基准,依次移除单个关键模块,并观察其对模型精确匹配率(EM)的影响,将 CLEAR-SQL 与设计的 4 个变体进行比较,结如表 4 所示。

Table 4 Ablation experiment results of CLEAR-SQL on the CSpider development set

表 4 CLEAR-SQL 在 CSpider 开发集上的消融实验结果 (%)

模型变体	精确匹配率	性能变化
CLEAR-SQL	73.4	-
w/o 语义对齐	60.2	-13.2
w/o 分层模式推理链	70.1	-3.3
w/o 混合对比检索	64.8	-8.2
w/o 移除执行反馈环路	71.3	-2.1

实验结果清晰地揭示了框架内各模块相辅相成且不可或缺的作用。当完全移除语义对齐机制、采用随机示例选择时,模型性能急剧下降了 13.2 个百分点,这充分证明了为 LLM 提供高质量、高相关的上下文示例是实现有效推理的根本前提。

此外,分层模式推理链模块的贡献最为显著(性能下降 8.6 个百分点),它通过引导 LLM 深度理解数据库模式,从根本上解决了跨语言场景中的语义与模式失配问题。进一步地,混合对比检索机制在语义检索的基础上,通过引入 SQL 结构信息,带来了额外的 3.3 个百分点的性能提升,凸显了在 Text-to-SQL 任务中结构对齐与语义对齐同等重要。

最后,执行反馈环路作为保障机制,通过迭代修正 LLM 的生成错误,贡献了 2.1 个百分点的性能增益,显著提升了框架的鲁棒性和在实际应用中的可用性。

3.5 跨语言上下文对齐有效性分析

为量化本文框架中混合对比检索模块在跨语言对齐方面的实际效果,设计了一项专项实验。该实验旨在分析不同检索策略下,检索到的示例 SQL 与标准答案 SQL 之间的结构骨架相似度。在 Multispider 基准的 7 种语言上,比较了 4 种不同的检索策略:①基线检索(R1),随机选择示例;②无掩码语义检索(R2),使用原始问题进行语义检索,不进行模式掩码;③跨语言语义检索(R3),采用本文框架中的语义对齐模块,对问题进行抽象化掩码后再进行语义检索;④混合结构—语义检索(R4),本文框架的完整方法,

结合了语义相似度与 SQL 结构相似度。

Table 5 SQL skeleton similarity analysis on MultiSpider

表 5 MultiSpider 的 SQL 骨架相似性分析 (%)								
检索策略	英语	德语	西班牙语	法语	日语	中文	越南语	平均值(6种语言)
单语训练(Monolingual Training)								
R1	33.4	29.1	27.9	28.3	26.5	28.7	25.8	28.5
R2	58.3	55.2	54.6	53.9	49.7	53.1	30.7	50.8
R3	77.2	69.6	69.3	67.7	67.0	72.6	42.5	66.6
R4	91.4	90.1	90.0	89.7	88.0	90.8	82.4	88.9
零样本直接预测(Zero-Shot Directly Predict)								
R1	33.4	29.9	28.5	29.1	27.8	30.2	26.4	29.3
R2	58.3	56.8	55.9	55.1	52.6	57.3	31.6	52.5
R3	77.2	74.6	76.1	76.0	72.2	77.9	43.8	71.1
R4	91.4	90.0	89.9	89.5	90.2	91.3	83.3	89.4

如表 5 所示,实验结果清晰地揭示了本文框架中各对齐策略的有效性。本文提出的混合结构—语义检索(R4)在零样本设置下取得了高达 88.9% 的平均结构相似度。相较于仅依赖语义对齐的纯语义检索(R3),其性能提升了 22.3 个百分点,这一巨大优势验证了在 Text-to-SQL 任务中引入 SQL 结构信息对于检索高质量示例至关重要。

此外,通过对比 R2(无掩码语义检索)和 R3(跨语言语义检索)可以发现,对查询进行抽象化掩码处理本身就能带来 15.8 个百分点的性能增益,这表明领域无关的抽象化处理能有效减少语言表层差异带来的噪声,从而实现更精准的跨语言语义匹配。

最后,在 R4 策略下,零样本设置(88.9%)与单语设置(89.4%)的平均结构相似度仅相差 0.5%。这一微小差距有力地证明了 CLEAR-SQL 框架设计的语言无关性,即无需目标语言的训练数据,也能达到与拥有该语言数据时几乎同等的对齐效果。

3.6 执行反馈机制有效性分析

为量化执行反馈环路在提升查询鲁棒性方面的实际贡献,本文在 CSpider 开发集上进行了一项专项分析。实验初始阶段,在未启用反馈环路的情况下,共识别出 32 个由模型生成的语法或模式错误,这些错误分布在 9 个不同的类别中。随后,启用执行反馈环路,对这些错误进行自动化的迭代修正。

错误校正具体结果如图 3 所示。分析表明,执行反馈环路成功修正了 32 个初始错误中的 21 个,实现了 65.6% 的总体错误校正率,有力地证明了该机制在保障 SQL 查询可用性方面的有效性。深入分析错误类型可以发现,在所有初始错误中,与模式链接(Schema Linking)相关的错误(如“表名错误”“列名错误”等)占比高达 78.1%(25/32),这证实了模式链接是跨语言 Text-to-SQL 任务中的核心挑战。

尽管挑战严峻,本文反馈环路在处理结构对齐问题上表现出色。例如,对于因复合键关系等导致的“表—列不匹配”错误,修正率达到了 80%(8/10),验证了基于 AST 的

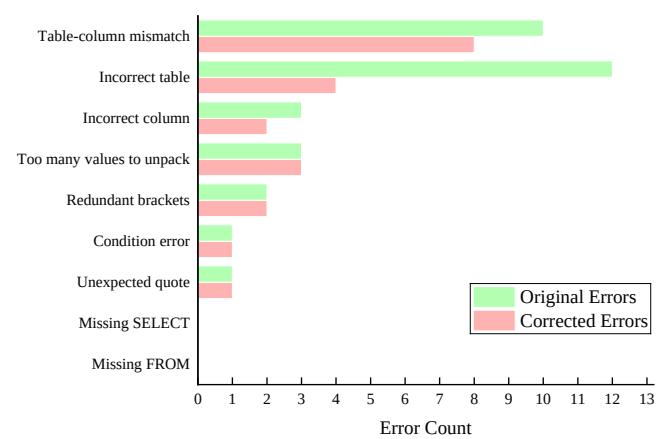


Fig. 2 Error correction analysis on CSpider-dev (total: 32 errors, corrected: 21)

图 2 CSpider-dev 上的纠错分析(共 32 个错误,已更正 21 个)

错误定位与结构化修正提示的精确性。

作为 CLEAR-SQL 框架的最后一道质量防线,执行反馈环路能够有效识别并修正大部分由 LLM 生成的语法和模式错误,极大地提升了框架鲁棒性和在实际应用中的可用性。

3.7 超参数敏感性研究

在混合对比检索模块中,通过一个可学习的权重超参数 α 将语义相似度 sim_{sem} 与结构相似度 sim_{struct} 进行线性组合,如式(5)所示。为确定该超参数的最优值,在 CSpider 开发集上进行了网格搜索实验。将参数空间定义为: $\alpha \in \{0.3, 0.4, 0.5, 0.6, 0.7\}$ 。

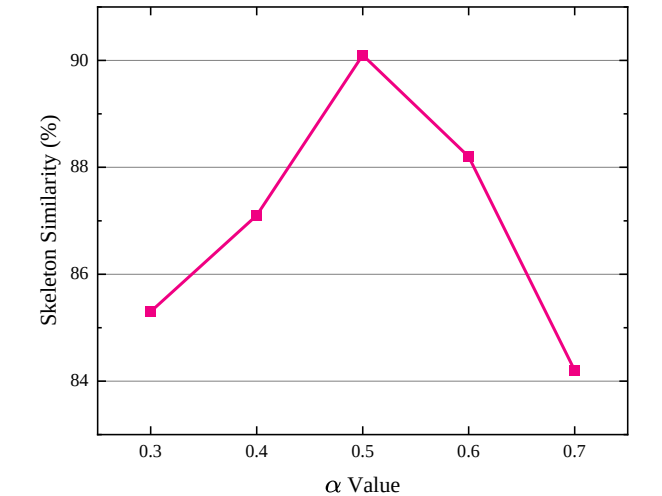


Fig. 3 Parameter sensitivity analysis based on SQL skeleton Jaccard similarity

图 3 基于 SQL 骨架 Jaccard 相似性的参数敏感度分析

实验结果如图 3 所示,图中横坐标为 α 的取值,纵坐标为检索到的示例与标准 SQL 之间的平均 Jaccard 相似度。可以清晰地看出,当 $\alpha=0.5$ 时,SQL 骨架的结构相似度达到峰值。这表明,在本文框架中,语义信息和结构信息同等重要,赋予它们相等的权重能够达到最佳检索效果。因

此,在本文所有主要实验中,将 α 的值都设定为0.5。

3.8 误差与局限性分析

为深入理解本文框架的性能瓶颈,对开发集上的错误案例进行了细致分析。尽管本文 CLEAR-SQL 框架在各项基准上均取得了显著性能提升,但其精确匹配率(EM)仍有提升空间。通过对错误案例的分析,总结出导致性能瓶颈的主要原因如下:

(1)模式链接复杂性。从消融实验(见表4)可以看出,分层模式推理链是贡献最大的模块,这反向证明了模式链接是 Text-to-SQL 任务中最核心的挑战。在面对包含数个表和数百个列的复杂数据库时,模型仍然可能在选择正确的表或列时出错,尤其是在存在多个看似合理的连接路径时。

(2)结构化推理偏差。部分查询不仅需要正确的表和列,还需要复杂的 SQL 结构(如嵌套子查询、GROUP BY 与 HAVING 的组合等)。虽然混合对比检索能够提升结构相似性,但 LLM 在零样本场景下生成特定复杂结构的能力仍有限,这是导致错误的重要原因之一。

(3)跨语言语义细微失真。在多语言场景下,尽管有语义抽象化,但某些语言的独特语法或表达习惯在翻译或对齐过程中仍可能导致细微的语义偏差,使得模型对 WHERE 子句中的约束条件理解错误。

这些从实验中观察到的具体问题,共同构成了本文框架当前性能的主要限制因素。

3.9 案例研究

为进一步验证本文框架在真实工业场景中的应用潜力和有效性,设计了一个模拟的企业级多语言销售数据查询案例。该案例的后台为一个包含 orders(订单)、products(产品)、customers(客户)、regions(区域)等数据表的复杂数据库。以一个典型、包含多重约束的中文业务查询为例,展示 CLEAR-SQL 的处理流程。

用户查询(中文):“帮我找出今年第三季度,在‘华南’大区购买了‘A 型号服务器’的客户里,订单总额最高的前三名客户的公司名称。”

CLEAR-SQL 处理流程展示如下:

(1)语义对齐与候选生成。系统首先将中文查询进行抽象化,语义抽象化后的结果为“帮我找出<col>的<col>,在<col>购买了‘<val>’的<tab>里,<tab>总额最高的前三名<tab>的名称。”并从英文示例库中检索出在查询意图上高度一致的 SQL 示例(即 ORDER BY...LIMIT 结构)的英文示例。

(2)分层模式推理。推理链模块分析查询后,准确识别出需要 orders, customers, products, regions 4 个核心表和 regions. name, orders. time, order. pay, customers. name, products. name 5 个核心列,并推断出它们之间基于 customer_id, product_id, region_id, orders_id 的正确连接路径。

(3)混合对比检索。通过结合语义与 SQL 结构相似

度,系统对检索到的示例进行重排序。得分最高的示例被优先选中。

(4)执行与反馈。LLM 在高质量示例的引导下,生成了初步的 SQL。在执行验证中,进行语法验证和模式一致性判断。LLM 据此生成了最终可正确执行的查询。

案例结论:该案例有力地证明了 CLEAR-SQL 不仅能处理学术基准,更能有效应对真实业务场景中语言异构(中一英)、查询逻辑复杂(多重约束与排名)、数据库模式复杂(多表连接)的挑战,展现了其作为多语言数据库自然语言接口的巨大工程应用潜力。

4 结语

本文提出了一个旨在解决跨语言 Text-to-SQL 核心挑战的创新框架——CLEAR-SQL,它通过集成混合对比检索、分层模式推理链和执行反馈循环,在不依赖“翻译—训练”范式的情况下,提升了 LLM 处理非英语查询的能力。

在 Spider、CSpider 和 Multispider 3 个基准上的广泛实验证明了 CLEAR-SQL 的卓越性能。与高度优化的单语 SOTA 基线相比,本文框架在 CSpider 上取得了超过 10 个百分点的性能提升;在 Multispider 的多种语言上,其平均精确匹配率(EM)同样显著优于传统微调方法。深入的消融实验进一步证实,本文提出的混合对比检索、分层模式推理链和执行反馈循环三大核心模块,对于提升解析准确性和鲁棒性均起到了关键且不可或缺的作用。

本文研究最重要的贡献在于,通过语言无关的框架设计,证明了在无需目标语言训练数据的情况下,实现高性能跨语言语义解析的可行性,为构建更智能、更易用的多语言数据库访问接口提供了坚实基础。

尽管本文框架在实验中表现出色,但仍存在一些根本性的局限。首先,框架性能受限于基座 LLM 本身能力,尽管 DeepSeek-R1 表现强大,但其在处理需要深度领域知识或复杂数值推理的查询时仍显不足,这是所有基于现有 LLM 的框架共同面临的挑战;其次,本文采用的 ICL 范式虽然灵活高效,但在面对需要大量上下文信息或极其专业的数据库时,其性能和成本效益可能会下降;最后,本文框架目前尚未覆盖对话式(Conversational)等多轮交互场景。

针对上述不足,未来研究可从如下方面展开:①探索模型轻量化技术,如知识蒸馏,以降低框架部署成本;②研究框架与知识图谱的融合,引入外部领域知识以增强模型推理深度;③探索更先进的强化学习策略以优化反馈修正过程;④将本文框架思想扩展至多语言对话式 Text-to-SQL 任务,以应对更复杂的真实世界交互场景。

参考文献:

- [1] LIU X, SHEN S, LI B, et al. A survey of NL2SQL with large language models: where are we, and where are we going?[DB/OL]. <https://arxiv.org/abs/2408.05109>.

- [2] ZHU X, LI Q, CUI L, et al. Large language model enhanced text-to-SQL generation: a survey[DB/OL]. <https://arxiv.org/abs/2410.06011>.
- [3] QIN B, HUI B, WANG L, et al. A survey on text-to-sql parsing: concepts, methods, and future directions [DB/OL]. <https://arxiv.org/abs/2208.13629>.
- [4] ACHIAM J, ADLER S, AGARWAL S, et al. GPT-4 technical report [DB/OL]. <https://arxiv.org/abs/2303.08774>.
- [5] YU T, ZHANG R, YANG K, et al. Spider: a large-scale human-labeled dataset for complex and cross-domain semantic parsing and text-to-sql task[DB/OL]. <https://arxiv.org/abs/1809.08887>.
- [6] SHERBORNE T, LAPATA M. Zero-shot cross-lingual semantic parsing [DB/OL]. <https://arxiv.org/abs/2104.07554>.
- [7] SHERBORNE T, XU Y, LAPATA M. Bootstrapping a crosslingual semantic parser[DB/OL]. <https://arxiv.org/abs/2004.02585>.
- [8] ZHANG Y, WANG J, WANG Z, et al. XSemPLR: cross-lingual semantic parsing in multiple natural languages and meaning representations [DB/OL]. <https://arxiv.org/abs/2306.04085>.
- [9] MOHAMMADJAFARI A, MAIDA A S, GOTTUMUKKALA R. From natural language to SQL: review of LLM-based Text-to-SQL systems [DB/OL]. <https://arxiv.org/abs/2410.01066>.
- [10] AWASTHI A, GUPTA N, SAMANTA B, et al. Bootstrapping multilingual semantic parsers using large language models [DB/OL]. <https://arxiv.org/abs/2210.07313>.
- [11] DO D T, NGUYEN M P, NGUYEN L M. Enhancing zero-shot multilingual semantic parsing: a framework leveraging large language models for data augmentation and advanced prompting techniques[J]. *Neurocomputing*, 2025, 618: 129108.
- [12] CHANG S, FOSLER-LUSSIER E. How to prompt LLMs for text-to-SQL: a study in zero-shot, single-domain, and cross-domain settings [DB/OL]. <https://arxiv.org/abs/2305.11853>.
- [13] LIU A, HU X, WEN L, et al. A comprehensive evaluation of ChatGPT's zero-shot text-to-SQL capability [DB/OL]. <https://arxiv.org/abs/2303.13547>.
- [14] NAN L, ZHAO Y, ZOU W, et al. Enhancing few-shot text-to-SQL capabilities of large language models: a study on prompt design strategies [DB/OL]. <https://arxiv.org/abs/2305.12586>.
- [15] SUN R, ARIK S Ö, MUZIO A, et al. SQL-PaLM: improved large language model adaptation for Text-to-SQL (extended) [DB/OL]. <https://arxiv.org/abs/2306.00739>.
- [16] SHI P, ZHANG R, BAI H, et al. XRICL: cross-lingual retrieval-augmented in-context learning for cross-lingual Text-to-SQL semantic parsing [DB/OL]. <https://arxiv.org/abs/2210.13693>.
- [17] TANWAR E, DUTTA S, BORTHAKUR M, et al. Multilingual LLMs are better cross-lingual in-context learners with alignment [DB/OL]. <https://arxiv.org/abs/2305.05940>.
- [18] DO D T, NGUYEN M P, NGUYEN L M. ZeLa: advancing zero-shot multilingual semantic parsing with large language models and chain-of-thought strategies[C]//Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024), 2024: 17783-17794.
- [19] DONG X, ZHANG C, GE Y, et al. C3: zero-shot text-to-sql with chatgpt [DB/OL]. <https://arxiv.org/abs/2307.07306>.
- [20] CHANG T A, TU Z, BERGEN B K. The geometry of multilingual language model representations [DB/OL]. <https://arxiv.org/abs/2205.10964>.
- [21] REIMERS N, GUREVYCH I. Sentence-bert: sentence embeddings using siamese bert-networks [DB/OL]. <https://arxiv.org/abs/1908.10084>.
- [22] MIN Q, SHI Y, ZHANG Y. A pilot study for Chinese SQL semantic parsing [DB/OL]. <https://arxiv.org/abs/1909.13293>.
- [23] DOU L, GAO Y, PAN M, et al. MultiSpider: towards benchmarking multilingual Text-to-SQL semantic parsing[C]//Proceedings of the AAAI Conference on Artificial Intelligence, 2023: 12745-12753.
- [24] GUO D, YANG D, ZHANG H, et al. Deepseek-R1: incentivizing reasoning capability in llms via reinforcement learning [DB/OL]. <https://arxiv.org/abs/2501.12948>.
- [25] GAO D, WANG H, LI Y, et al. Text-to-SQL empowered by large language models: a benchmark evaluation [DB/OL]. <https://arxiv.org/abs/2308.15363>.
- [26] LI J, HUI B, CHENG R, et al. Graphix-T5: mixing pre-trained transformers with graph-aware layers for Text-to-SQL parsing[C]//Proceedings of the AAAI Conference on Artificial Intelligence, 2023: 13076-13084.
- [27] ZHAO Y, JIANG J, HU Y, et al. Importance of synthesizing high-quality data for text-to-sql parsing [DB/OL]. <https://arxiv.org/abs/2212.08785>.
- [28] VOUGIOUKLIS P, PAPASARANTOPOULOS N, ZHENG D, et al. Fasttrat: fast and efficient cross-lingual text-to-sql semantic parsing [C]//Proceedings of the 13th International Joint Conference on Natural Language Processing and the 3rd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics, 2023: 564-576.
- [29] WANG B, SHIN R, LIU X, et al. RAT-SQL: relation-aware schema encoding and linking for text-to-sql parsers [DB/OL]. <https://arxiv.org/abs/1911.04942>.
- [30] CONNEAU A, KHANDELWAL K, GOYAL N, et al. Unsupervised cross-lingual representation learning at scale [DB/OL]. <https://arxiv.org/abs/1911.02116>.
- [31] LIU Y, GU J, GOYAL N, et al. Multilingual denoising pre-training for neural machine translation[J]. *Transactions of the Association for Computational Linguistics*, 2020, 8: 726-742.
- [32] POURREZA M, LI H, SUN R, et al. CHASE-SQL: multi-path reasoning and preference optimized candidate selection in text-to-sql [DB/OL]. <https://arxiv.org/abs/2410.01943>.
- [33] GAO Y, LIU Y, LI X, et al. XiYan-SQL: a multi-generator ensemble framework for Text-to-SQL [DB/OL]. <https://arxiv.org/abs/2411.08599>.

(责任编辑:孙 娟)