计算机系统应用 ISSN 1003-3254, CODEN CSAOBN
Computer Systems & Applications [doi: 10.15888/j.cnki.csa.009356]

E-mail: csa@iscas.ac.cn
http://www.c-s-a.org.cn
Tel: +86-10-62661041

基于中文自然语言的 SQL 生成综述①

郑耀东, 李旭峰, 陈和平, 贺桂娇

(广州软件学院 计算机系, 广州 510990)
通信作者: 郑耀东, E-mail: zyd@mail.seig.edu.cn

摘　要: 自然语言转为 SQL (NL2SQL) 的研究有较高的应用价值, 随着深度学习技术的成熟, 越来越多的研究者开始将深度学习技术应用于 NL2SQL 任务中. 本文梳理了英文和中文领域 NL2SQL 的研究现状, 总结按年份发布的数据集和模型, 对比当前 4 大中文 NL2SQL 数据集的特点, 阐述了当前基于深度学习的 NL2SQL 任务的基本框架以及针对中文领域的单表简单问题和跨表复杂问题所适用的典型模型, 介绍了一般常用的模型评测方法, 并提出未来研究方向的展望.

关键词: NL2SQL; 深度学习; 中文数据集; 自然语言处理

引用格式: 郑耀东,李旭峰,陈和平,贺桂娇.基于中文自然语言的 SQL 生成综述.计算机系统应用. http://www.c-s-a.org.cn/1003-3254/9356.html

Survey on SQL Generation Based on Chinese Natural Language

ZHENG Yao-Dong, LI Xu-Feng, CHEN He-Ping, HE Gui-Jiao

(Department of Computer Science, Software Engineering Institute of Guangzhou, Guangzhou 510990, China)

Abstract: The research on natural language to SQL (NL2SQL) has high application value. With the maturity of deep learning technology, increasingly more researchers have begun to apply deep learning technology to NL2SQL tasks. This study reviews the research status of NL2SQL in English and Chinese fields and summarizes the datasets and models published by year. Additionally, it compares the characteristics of the four major Chinese NL2SQL datasets and expounds on the basic framework of NL2SQL tasks based on deep learning and typical models for simple single-table problems and complex cross-table problems in Chinese NL2SQL fields. Finally, the commonly adopted model evaluation methods are introduced, and future research directions are put forward.

Key words: natural language to SQL (NL2SQL); deep leaning; Chinese dataset; natural language processing (NLP)

1　引言

随着现代信息技术的发展以及数据的海量式增长, 人们希望以更自然、便捷的方式从数据库中获取信息, NL2SQL (natural language to SQL) 也称为 Text2SQL (text to SQL) 应运而生, 旨在将用户熟悉的自然语言转换为 SQL 序列, 继而完成数据库查询工作, 从而打破人与结构化数据库之间的壁垒.

NL2SQL 问题可表示为: 给定自然语言查询问句 Q 以及相应的数据库表 T, 由 NL2SQL 模型生成 SQL 语句, 其公式如式 (1) 所示:

$$f(Q,T) \to \text{SQL} \tag{1}$$

早期的 NL2SQL 任务大多基于规则和流水线, 所采用的数据集也是某个领域的简单数据集, 研究方法难以复制. 近几年, 随着 WikiSQL 和 Spider 等大规模人工标注数据集的发布, 以及深度学习和自然语言处理技术的快速发展, NL2SQL 领域的研究受到广泛关注. 研究者提出了一系列的新方法, 不断地推进着这一

① 基金项目: 广州软件学院科研项目 (ky202113)
收稿时间: 2023-06-12; 修改时间: 2023-07-19; 采用时间: 2023-08-17; csa 在线出版时间: 2023-10-20

领域的研究进展. 本文对 NL2SQL 研究工作进行综述, 并重点侧重于中文领域的研究. 首先梳理了英文和中文领域 NL2SQL 的研究现状, 总结了主要的方法和典型模型. 接着总结按年份发布的数据集和模型, 对比当前 4 大中文 NL2SQL 数据集的特点. 然后本文阐述了当前基于深度学习的 NL2SQL 任务的基本框架以及针对中文领域的单表简单问题和跨表复杂问题所适用的典型模型, 并介绍了一般常用的模型评测方法. 最后本文提出未来研究方向的展望.

2 NL2SQL 研究现状

2.1 英文领域现状

国外开展 NL2SQL 较早, 早期传统的研究大多通过对特定数据库人工制定匹配规则的方式来完成. 如 LUNAR[1]、LADDER[2]、Chat-80[3]、PRECISE[4] 等. 接着出现了针对数据集的研究, 例如 ATIS[5]、GeoQuery[6]、JOBS[6]、Scholar[7]、Academic[8] 等. 但针对这些数据集的研究并没有后来的 WikiSQL 或 Spider 那样得到广泛的应用. 由于这些数据集仅关注单一领域, 而且样本数太少, 所提出的研究方法即使在特定领域表现良好, 但通常不具备泛化能力和通用性. 2017 年 Zhong 等人提出了第 1 个大规模人工标注的 NL2SQL 数据集 WikiSQL[9], 研究者们开始基于 WikiSQL 数据集使用深度学习方法来解决 NL2SQL 方面的问题. 对于 WikSQL 的单表查询任务, 基于草图和槽位填充的方法比较有效, 典型的模型包括 SQLNet[10]、TypeSQL[11]、STAMP[12]、Coarse2Fine[13]、IncSQL[14] 等.

随着深度学习的发展, Transformer 和 BERT 模型相继被提出并在自然语言处理任务上取得优异的成绩. Hwang 等人于 2019 年提出 SQLova 模型[15], 第 1 次把大规模预训练模型应用于 NL2SQL 任务. SQLova 使用 BERT 作为模型的输入表示层, 以此取代 SQLNet 中的词向量表示. SQLova 将预测 SQL 语句任务划分为 6 个子任务, 并将自然语言问句和数据库表的列名均作为网络的输入进行编码. 大规模预训练模型 BERT 的引入, 大幅提升了模型的语义分析能力, 模型取得惊人的效果. He 等人提出 X-SQL 模型[16], 在预训练模型的选择上, 采用了 MT-DNN 来代替 BERT, 并在对列名编码时添加了上下文注意力机制, 取得更好的效果. 2020 年, Lyu 等人提出 HydraNet 模型[17], 没有将所有列名和问句连接在一起做 BERT 编码, 而是将每个列名分别与问句做编码, 再通过规则按列输出组装成 SQL 查询, 取得更好的效果. 2021 年, Xu 等人提出 SeaD 模型[18], 没有采用基于草图和槽填充的方法, 而是直接训练序列到序列 (Seq2Seq) 的模型, 使用 Transformer 作为基础架构, 采用修改和删除列名以及打乱实体顺序的方式做数据增强, 并使用一种条件敏感的执行指导策略来进行解码, 从而最大化序列生成方法的潜能, 在 WikiSQL 数据集上取得了最好成绩. 总结 WikiSQL 数据集主要模型的贡献和效果如表 1 所示.

表 1 WikiSQl 数据集主要模型的贡献和执行准确率

模型名称	提出年份	模型主要贡献	执行准确率 (%)	
			验证集	测试集
SeaD	2021	用新的模式感知去噪目标来训练模型, 以提高NL2SQL任务的Seq2Seq生成性能	92.9	93
HydraNet	2020	每次编码NLQ和一列, 将解码任务分为与具体列有关和无关的两类	89.1	89.2
X-SQL	2019	在预训练模型基础上, 增加上下文增强层增强模式表示, 提出一种基于KL散度的目标函数解决子任务分别优化的问题	89.5	88.7
SQLova	2019	基于草图, 首次使用预训练模型BERT加强数据库和NLQ的表示	87.2	86.2
IncSQL	2018	使用序列到动作的解码模型, 保证了解码的上下文依赖	84	83.7
Coarse2Fine	2018	提出了一种从粗到精的语义解析解码框架, 分阶段生成where子句	79.0	78.5
TypeSQL	2018	利用列类型和schema link等先验知识辅助编码过程, 使模型更好地理解NLQ中稀有实体和数字	74.5	73.5
Seq2SQL	2017	将列名、NLQ和SQL关键字一起编码, 设计3个子任务, where子句引入强化学习	60.8	59.4
SQLNet	2017	基于草图和槽填充思想, 将NL2SQL任务转化为6个子任务	69.8	68

以上从 2017 年开展的基于深度学习的研究都是针对 WikiSQL 数据集, 该数据集仅针对某个领域单个表做简单查询. Yu 等人于 2018 年提出更加复杂的 Spider 数据集[19], 包含 oder by、group by、join 以及嵌套查询. 由于 SQL 更加复杂, 以往针对 WikiSQL 数据集的模型不再适用. 研究者们先后提出了 SyntaxSQLNet[20]、RCSQL[21]、IRNet[22]、RYANSQL[23]、SmBoP[24]、BRIDGE[25]、DT-Fixup[26] 等模型, 采用语法树、schema

linking 等技术提升模型效果.

随着图神经网络 (graph neural network, GNN) 在深度学习领域表现出优异的性能, 越来越多的研究者开始在 NL2SQL 任务中利用图神经网络来表示关系型数据库的数据结构, 以提升数据库信息的利用效率. GNN[27]、Global-GNN[27]、RAT-SQL[28]、Shadow-GNN[29]、LGESQL[30]、S^{2}SQL[31] 、INSL[32] 等模型相继被提出, 在 Spider 数据集上表现优异.

SParC[33] 和 CoSQL[34] 数据集是对 Spider 数据集的扩展, 具有上下文语义相关、跨域、多轮对话等特性, 与实际的问答场景更加接近, 新的数据集为该领域的研究带来更大的挑战. EditSQL[35]、Bertrand-DR[36]、HIE-SQL[37]、UniSAr[38]、RASAT[39]、CQR-SQL[40] 等模型增强 SQL 在语境理解方面的解析能力, 都取得了相对较好的效果.

随着模型参数规模的增加, 大语言模型 (LLM) 展现出涌现能力, Codex、chatGPT、GPT-4、PaLM、OPT 等大语言模型在零样本、小样本或上下文学习方面取得显著的成就. Pourreza 等人提出 DIN-SQL 模型[41], Sun 等人提出 SQL-PaLM 模型[42], 在 Spider 数据集上都取得了最优成绩.

Spider 数据集更加贴近实际应用场景, 近几年的绝大部分工作都在 Spider 数据集上进行测试, 由于难度较大, 在 Spider 数据集上模型的准确率要低于 WikiSQL 数据集, 总结主要模型的贡献和效果如表 2 所示.

英文 NL2SQL 的研究已经比较成熟, 总结英文领域 NL2SQL 的主要方法和典型模型如表 3 所示.

表 2　Spider 数据集主要模型的贡献和执行准确率

模型名称	提出年份	模型主要贡献	执行准确率 (%)	
			验证集	测试集
SQL-PaLM	2023	应用大语言模型PaLM2的few-shot和微调能力	—	78.2
DIN-SQL+GPT-4	2023	应用大语言模型GPT-4	—	74.2
S²SQL+ELECTRA	2021	将语法注入到question-schema图编码器, 有效利用问句的语法依赖信息来提高性能	76.4	72.1
LGESQL+ELECTRA	2021	通过引入线性有向图, 在简化图的同时突出边信息	75.1	72
RASAT+PICARD	2022	Transformer Seq2Seq架构, 增强了关系感知自注意力	75.3	70.9
ShadowGNN+RoBERTa	2020	通过忽略数据库中语义项目的名称, 在图投影神经网络中利用抽象模式来获得问题和模式的非词汇化表示	72.3	66.1
LGESQL+GloVe	2021	通过引入线性有向图, 在简化图的同时突出边信息	67.6	62.8
SmBoP+BART	2020	在解码步骤构建前k个子树, 并允许自底向上解码特定高度的子树	66	60.5
BRIDGE+BERT	2020	用标记序列表示问句和数据库schema, 其中字段的子集用问句中的单元格值进行扩充, 以解决跨数据库场景下问句和数据库的依赖关系	65.5	59.2
RAT-SQL	2019	引入了数据库模式图 (schema graph)来表示数据库结构	60.6	53.7
Global-GNN	2019	在GNN基础上, 全局选择与问题相关的列和表, 对候选结果进行重排	52.7	47.4
IRNet	2019	设计中间表示语言SemQL, 可有效解决复杂SQL的生成问题	53.2	46.7
GNN	2019	数据库结构用图神经网络建模	40.7	39.4
EditSQL	2019	提出查询编辑机制, 通过交互的方法来提高生成 SQL 的准确率	36.4	32.9
RCSQL	2019	基于自注意力机制的数据库模式编码器, 以递归方式支持嵌套查询	28.5	24.3
SyntaxSQLNet	2018	基于模版的方法, 语法树	18.9	19.7
SQLNet	2017	基于草图和槽填充思想, 将NL2SQL任务转化为6个子任务	10.9	12.4
TypeSQL	2018	利用列类型和schema link等先验知识辅助编码过程, 使模型更好地理解NLQ中稀有实体和数字	8	8.2

表 3　英文 NL2SQL 方法和典型模型

方法	典型模型
基于规则和流水线	LUNAR、LADDER、Chat-80、PRECISE、Nchiql
基于草图	SQLNet、TypeSQL、STAMP、Coarse2Fine、IncSQL、X-SQL、SQLova、RYANSQL、SyntaxSQLNet
多轮对话	EditSQL、HIE-SQL、UniSAr、RASAT、CQR-SQL
基于图神经网络	GNN、Global-GNN、RAT-SQL、ShadowGNN、LGESQL
基于大语言模型	DIN-SQL、SQL-PaLM

2.2 中文领域现状

中文领域 NL2SQL 发展相对落后, 一方面, 中文语法和语义相对复杂, 相比于英文, 处理难度较大. 另一方面, 早期中文 NL2SQL 数据集的缺乏也在一定程度上限制了中文问答系统的发展. 李保利等人于 1999 年提出了 WTCDIS 接口[43], 能够处理省略及乱序的用户输入, 但系统的可移植性差, 不能直接迁移到其他领域上. 孟小峰等人提出 Nchiql[44], 对自然语言问句进行语法和语义分析, 首先将自然语言问句表示为语义依存树, 然后将语义依存树转变为 SQL 查询. Shen 等人提出 SPSQL[45], 结合流水线方法和深度学习的方法, 将任务分解为表选择、列选择、SQL 生成和值调整 4 个子任务, 有效地提升了性能.

2019 年, 西湖大学 Min 等人将 Spider 数据集中的自然语言问句部分进行了中文翻译, 发布了 CSpider[46] 中文复杂 SQL 数据集. 通过这个数据集应用 Yu 等人[20] 的 SyntaxSQLNet 模型作为基线系统的测试, 进一步探索了中文 NL2SQL 任务的挑战.

2019 年 6 月, 追一科技举办了首届中文 NL2SQL 比赛, 这让中文 NL2SQL 受到广泛关注. 本次比赛还发布了首个大规模中文 NL2SQL 数据集 TableQA[47], 以金融和通用领域数据为数据源, 方便后续研究者展开研究. 主办方给出一个应用 SQLNet 模型的基线, SQLNet 针对英文 WikiSQL 数据集效果较好, 但在应用中文数据集时效果不佳. 主要的挑战包括: where 条件值预测不够精准; 问句中没有提到 SQL 语句中的选择列和条件列; 问句和条件关系表示不一致等. 另外, 研究者在 TableQA 数据集上直接应用 SQLova 模型得到的效果虽然好于 SQLNet, 但还有比较大的改善空间. Zhang 等人在本次比赛提出的 M-SQL 模型[48], 以哈工大讯飞联合实验室提出的中文 BERT 预训练模型作为编码器作初始化参数, 搭建了多任务学习框架, 通过联合学习提高了模型的准确度, 获得了比赛的第一名. F-SQL[49] 在 M-SQL 的基础上, 使用门机制融合数据库模式和表内容, 有效地增强了列表示, 提高了 SQL 查询生成性能.

2020 年, 百度举行语言与智能技术竞赛, 此次比赛提供大规模开放域的复杂中文 NL2SQL 数据集 DuSQL[50]. 随数据集发布了 IRNet-Ext 模型, 该模型为 IRNet 模型的扩展版本, 以适应中文 DuSQL 数据集的特点.

2021 年, 微软研究院联合西安交大、北航等机构发布了 CHASE 数据集[51], 通过人工标注和审核丰富了问题的多样性和难度, 也更加贴近实际应用场景. 基于多轮对话的模型, 如 UniSAr, 除了适用于 WikiSQL、Spider 等英文数据集, 在中文数据集 DuSQL 和 CHASE 上也取得了较好的实验效果.

汇总中文 NL2SQL 模型和主要贡献如表 4 所示.

总结 NL2SQL 领域按年份发布的中、英文数据集和模型汇总如图 1 所示.

表 4 中文 NL2SQL 模型和主要贡献

模型名称	数据集	提出年份	模型主要贡献	测试集执行准确率 (%)
SPSQL	电力行业数据	2023	结合流水线方法和深度学习的方法, 将任务分解为表选择、列选择、SQL生成和值调整4个子任务	95.6
UniSAr	CHASE	2022	提出一种统一结构感知自回归语言模型, 可以在一个统一的框架中解决跨各种设置的NL2SQL的问题	42.2
EditSQL	CHASE	2021	随CHASE发布的基线模型	37.7
IRNet-Ext	DuSQL	2020	针对中文语言的特点, 在IRNet模型基础上做改进	54.3
F-SQL	TableQA	2020	在M-SQL的基础之上, 使用门机制融合数据库模式和表内容, 有效地增强了列表示	91.8
M-SQL	TableQA	2020	以中文BERT-wwm-ext模型作为预训练, 搭建多任务学习框架	90.2
SQLova	TableQA	2020	基于草图和中文BERT模型, 在中文数据集上表现更好	49.7
SQLNet	TableQA	2020	首次发布中文NL2SQL数据集, 应用SQLNet模型作为基线	30.1
SyntaxSQLNet	CSpider	2019	翻译英文Spider数据集, 应用SyntaxSQLNet模型作为基线	14.1
Nchiql	—	2001	对自然语言问句进行语法和语义分析, 将自然语言问句表示为语义依存树	—
WTCDIS	—	1999	数据库汉语言查询接口, 能够处理省略及乱序的用户输入	—

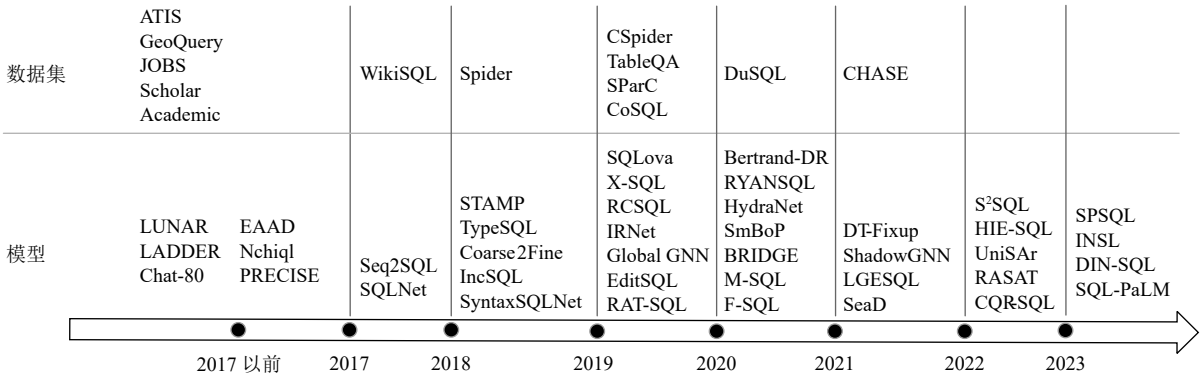

图 1 NL2SQL 数据集和模型汇总

3 中文 NL2SQL 数据集

随着近几年深度学习技术在自然语言处理领域的广泛应用, NL2SQL 任务取得了较大进展, 而现阶段对 NL2SQL 任务的研究大部分还是基于英文数据集. 对于中文 NL2SQL 技术的研究还处在初级阶段, 相对于英文, 中文表达更加多样化, 机器理解更加困难. TableQA 数据集为中文领域公布的第 1 个大型 NL2SQL 数据集, 由追一科技公司于 2019 年天池大赛中发布. 该数据集包含金融以及通用领域的用户自然语言查询文本与 SQL 语句的标注匹配数据对, 包含 64 891 个问题和相对应的 20 311 个唯一 SQL 语句. 和 WikiSQL 数据集类似, TableQA 中的 SQL 语句仅适用于单个表, 适用的范围较窄, 无法进行多表 SQL 查询生成.

CSpider 数据集为 Spider 数据集的中文翻译, 由 9691 个问句和 5 263 个复杂 SQL 查询组成, 涉及 166 个数据库, 880 个表涵盖 138 个不同的域. 作者翻译了问句和数据库的内容, 但保留了英文的数据表名和数据列名以方便工程应用, 这也增加了问句和数据库对应的难度. 另外, 在翻译的过程中, 对一些英文的地名和人名进行了本地化处理. CSpider 根据 SQL 语句中出现关键字的数量、选择和条件的复杂度以及 group by、order by 和嵌套查询、聚合操作的应用等为依据将每个问题分为了简单、中等、困难、极难 4 个等级, 方便更好地验证模型在不同查询上的性能.

百度 2019 年在语言与智能技术竞赛中发布的 DuSQL 是一个跨域的面向实际应用的数据集. 包含 200 个数据库、覆盖 164 个领域的 813 个表以及 23 797 个问句/SQL 对. 数据源来自百度百科、贴吧以及权威网站等, 该数据集更贴近真实应用场景, 问题覆盖了匹配、计算、推理等实际应用中的常见形式.

2021 年发布的 CHASE 数据集是一个复杂的高质量多轮交互数据集. 它由 280 个不同领域的数据库和 5 459 个问题序列以及 17 940 个问句/SQL 对组成. CHASE 中的每个问题都有丰富的语义注释, 包括其 SQL 查询、上下文依赖和模式链接. 作为第一个跨域的多轮 NL2SQL 中文数据集, 作者翻译了英文多轮对话数据集 SParC, 并做了修复和优化以组成 CHASE-T. CHASE-C 精选 DuSQL 中 120 个高质量数据库, 包括体育、教育、娱乐等 60 个子领域, 并做了人工审核和优化.

NL2SQL 数据集有不同的分类方法, 根据覆盖领域的数量, 可以分为单领域和多领域; 根据每个数据库包含表的数量, 可分为单表和跨表; 根据交互方式, 可分为单轮和多轮. 表 5 所示为 4 个中文数据集的基本信息和分类.

表 5 中文 NL2SQL 数据集的基本信息和分类

数据集	发布机构	发布年份	问句/SQL对	数据库个数	表个数	涉及领域	交互方式	单表/跨表
TableQA	追一科技	2019	20 311	5 291	5 291	多领域	单轮	单表
CSpider	西湖大学	2019	9 691	166	876	多领域	单轮	跨表
DuSQL	百度	2020	23 797	200	813	多领域	单轮	跨表
CHASE	微软研究院、西安交大、北航	2021	17 940	280	1 280	多领域	多轮	跨表

NL2SQL 数据集的发展趋势是从单表、单领域、单轮、简单问题到跨表、多领域、多轮、复杂问题, 任务难度也逐步提升. 中文领域的 TableQA 到 CHASE, 其难度也逐步加深, 数据样例如表 6 所示.

表 6 中文 NL2SQL 数据集样例

数据集	问题/SQL对样例
TableQA	问句: 净资产收益率达到20以上或者季度每股盈余达到1以上的有哪些证券? SQL: select 证券代码 from 国内钢企ROE及PE表 where ROE > 20 or EPS > 1
CSpider	问句: 1998参加全明星赛的球员的名字和姓氏是什么? SQL: select name_first, name_last from player as T1 join all_star as T2 on T1.player_id = T2.player_id where year = 1998
DuSQL	问句: 在年龄不到26岁获得欧洲杯金球奖的足球运动员中, 给出获得金球奖不止5个的足球运动员的中文名字及其国籍 SQL: select T2.中文名 , T2.国籍 from 欧洲杯金球奖 as T1 join 足球运动员 as T2 on 欧洲杯金球奖.运动员id == 足球运动员.词条id where T2.年龄 < 26 group by T1.运动员id having COUNT(*) > 5
CHASE	问句: 2019年哪个省份的生产总值最高? SQL: select T2.名称 from 省生产总值 as T1 join 省份 as T2 on T1.省份id = T2.省份id where T1.年份 = 2019 order by T1.GDP DESC LIMIT 1 问句: 2015年的呢? SQL: select T2.名称 from 省生产总值 as T1 join 省份 as T2 on T1.省份id = T2.省份id where T1.年份 = 2015 order by T1.GDP DESC LIMIT 1

4 中文 NL2SQL 典型模型和评测方法

4.1 模型基本框架

基于深度学习的研究方法, NL2SQL 的任务可被认为是一个类似机器翻译的序列到序列的生成任务, 主要采用 Encoder-Decoder 框架, 利用端到端的模型, 实现将自然语言问句翻译为结构化的 SQL 语句序列, 如图 2 所示. 首先将自然语言问句和数据库表结构通过编码器编码为向量, 再通过解码器解码后输出 SQL 语句.

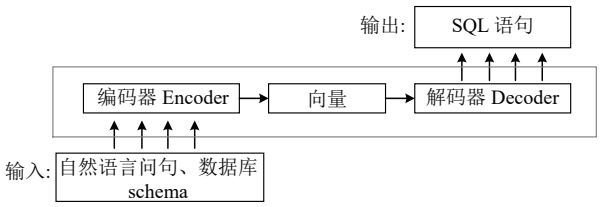

图 2 NL2SQL 任务 Encoder-Decoder 框架

4.2 单表简单模型

单表查询的问答是实际场景中应用十分广泛的一项 NL2SQL 任务, 针对英文 WikiSQL 数据集的研究已经十分成熟. 主流的方案采用槽位填充的方法, 可以充分利用 SQL 语句的语法规则. SQL 语句具有固定的语法格式, 按顺序, 它由 select、where、and/or 关键字组成, 后面跟着要填充的内容. 如图 3 所示为 SQLNet 模型 , $agg 表示聚合函数、$column 表示列名、$op 表示运算符、$value 表示值, “*”表示该部分出现多次. 它将 NL2SQL 任务转化为 6 个子任务: select-column (列选择)、select-agg (列聚合)、where-number (条件数)、where-column (筛选条件对应列)、where-op (条件操作符) 以及 where-value (条件值). SQLNet 为解耦出的每个子任务依据各自任务目标设计了不同输出层. select-column 的输出层对所有列执行分类任务, select-agg 的输出层对聚合函数 (如 MAX、AVG、SUM、MIN、COUNT) 执行分类任务, 子任务间存在依赖关系.

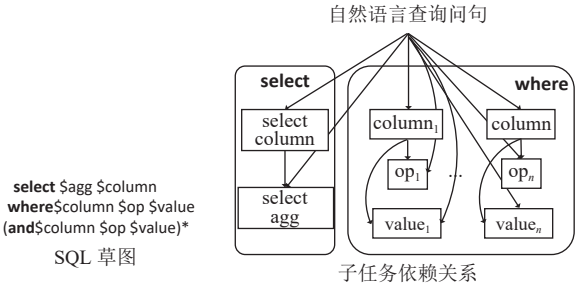

图 3 SQLNet 模型

和 SQLNet 类似, 基于草图的研究都使用了类似的任务划分, 比较知名的有 SQLova 和 X-SQL, 这两个模型引入了预训练模型 BERT, 效果更好并基本达到了 WikiSQL 数据集的极限. 而当研究者把 SQLova、X-SQL 等模型直接应用在中文 TableQA 数据集上时, 发现效果并理想. TableQA 比 WikiSQL 更加复杂, 需要添加两个额外的子任务: 一个用于预测所选列的数量, 另一个用于预测 where 子句中的条件关系. 另外如果查询中存在多个值, 并且这些值属于不同的列时, 用 SQLova 和 X-SQL 无法准确提取. 而且, TableQA 中文的查询形式比较随意, 问句中可能不会出现数据库的

内容. Zhang 等人[48] 扩展了 WikiSQL 模型框架, 提出了 M-SQL. M-SQL 包含 8 个子模型, 分别是 S-num (列数量)、S-col (列选择)、S-col-agg (列聚合)、W-num-op (条件列的关系和个数)、W-col (条件列)、W-col-op (条件列操作)、W-col-val (条件列的值) 和 W-cal-match (匹配条件列的值), 其草图如图 4 所示.

另外, 针对传统模型无法准确提取包含多个值和多个列的样本的值的问题, M-SQL 改进了提取方法, 将基于列的值提取分为两个模块: 值提取和值列匹配. M-SQL 的基本框架是多任务学习架构. 其整体架构如图 5 所示. 总体模型由 3 部分组成, 编码器层、表示层和子模型层. 编码器使用了哈工大讯飞联合实验室提出的中文 BERT-wwm-ext 模型, 和中文预训练模型 ERNIE 类似, 采用全词掩码策略, 可以更好地学习中文词向量表示. M-SQL 模型在处理中文单表简单查询任务中表现优异, 可看作是当前中文 NL2SQL 单表查询任务的最佳方案.

```
select($agg$column)*
where$wop($column$op$value)*
```

图 4 M-SQL 模型 SQL 草图

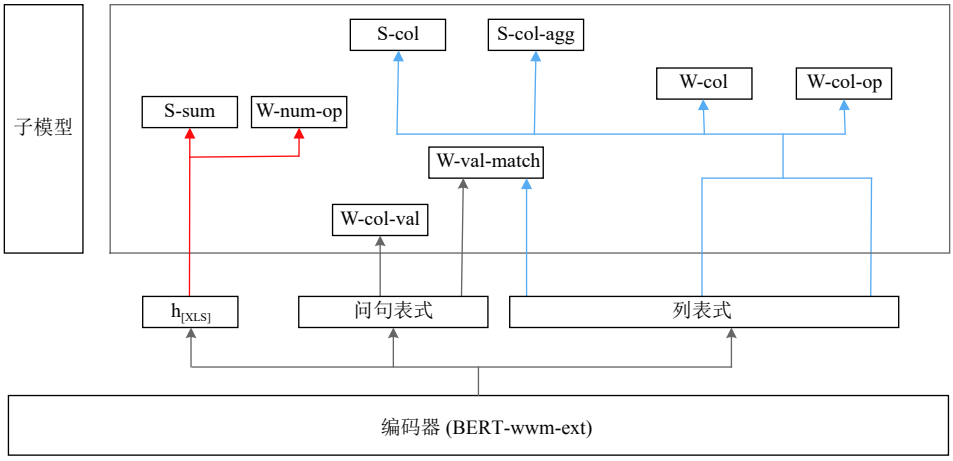

图 5 M-SQL 模型架构

4.3 跨表复杂模型

与 NL2SQL 单表任务相比, 多表任务数据集的整理要复杂很多, 而且所对应的 SQL 查询形式也多种多样, 涉及多关键词组合、嵌套、多表连接等. IRNet 设计了一种中间表示语言 SemQL, 可以有效地解决复杂 SQL 的生成问题. 如图 6 所示, SemQL 可看作是一个中间语言, 承接了自然语言和 SQL 的连接, 自然语言先生成 SemQL 的语法树, 再通过对语法树的解析则可转换为 SQL.

RAT-SQL 做 schema linking 的方法和 IRNet 类似, 不同的是 RAT-SQL 中引入了数据库模式图 (schema graph) 来表示数据库的结构. 一个数据库包含多个表, 表间通过主、外键连接, 把表名和列看作节点的话, 一个数据库即可构成了一张图的形式.

RAT-SQL 模型将 schema linking 和 schema graph 的信息嵌入 Transformer, Transformer 结构可看作是一系列自注意力层的叠加, 编码时自然语言问句和表名、列名等 schema 融合一起作为输入, 构建 relation embedding. 解码时, 通过树形解码器, 生成 SQL 对应的抽象语法树, 最后通过解析抽象语法树来生成最终的 SQL. 编码和解码过程如图 7 所示.

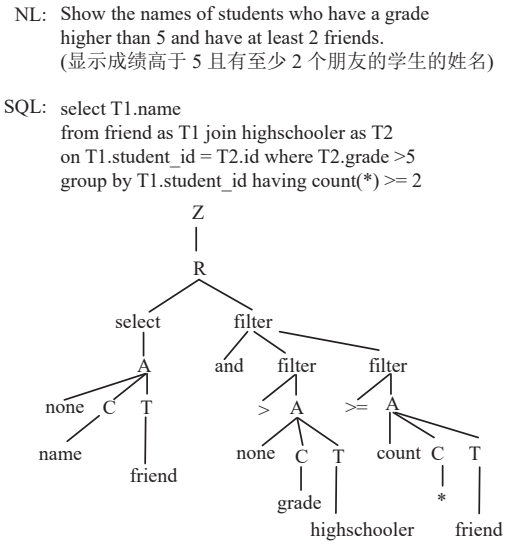

图 6 SemQL 示例

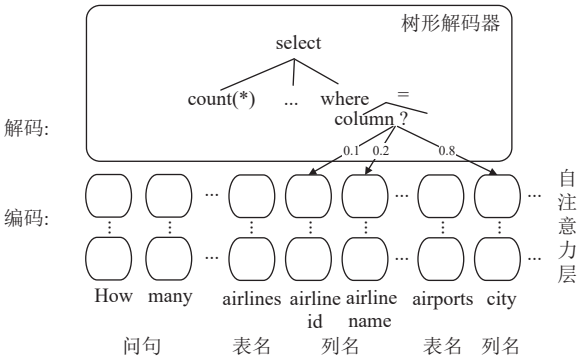

图 7 RAT-SQL 编码和解码

当前无论英文 Spider 还是中文 CSpider 和 DuSQL 数据集, 主流的方案都是基于 RAT-SQL 或者相关的改进模型, 如 S^2SQL、LGESQL 等. 新的模型在语义理解、关系发现、编码解码能力方面都追求更好的表现.

4.4 评测方法

NL2SQL 任务的评测方法主要包括逻辑形式准确率 (logical form accuracy) Acc_{lf}、执行准确率 (execution accuracy) Acc_{ex}、查询匹配准确率 (query-match accuracy) Acc_{qm} 等.

逻辑形式准确率是指通过模型所生成的 SQL 语句与正确的 SQL 语句是否完全一致, 包括 select 是否一致、agg 是否一致、conds 是否一致、value 是否一致等. 式 (2) 为逻辑形式准确率的计算公式, 其中 N_{lf} 是逻辑形式上完全正确的样本个数, N 是样本总数. 该评估方式的缺点是会过滤掉仅是由于 select 列的顺序不同或者条件的顺序不同但执行结果是正确的情况, 因此会导致准确率虚低.

$$Acc_{\text{lf}} = \frac{N_{\text{lf}}}{N} \tag{2}$$

执行准确率是指执行预测的 SQL 语句, 数据库返回正确结果的样本数占比. 其公式如式 (3) 所示, 其中 N_{ex} 是执行结果预测正确的样本个数, N 是样本总数. 执行准确率只要求最终由数据库执行引擎返回的结果与正确 SQL 语句所执行的结果一致, 但无法保证生成的 SQL 是语义正确的, 因此会导致准确率虚高.

$$Acc_{\text{ex}} = \frac{N_{\text{ex}}}{N} \tag{3}$$

查询匹配准确率是指模型生成的 SQL 和正确的 SQL 都以标准形式表示, 再计算两者的匹配精度. 其公式如式 (4) 所示, 其中 N_{qm} 是标准化后的预测的 SQL 和正确 SQL 匹配的样本个数, N 是样本总数.

$$Acc_{\text{qm}} = \frac{N_{\text{qm}}}{N} \tag{4}$$

实际应用中, 针对 SQL 子句的评估, 也经常采用拆分后做逻辑形式准确率评估的方法, 如 $Acc_{\text{S-agg}}$、$Acc_{\text{S-col}}$、$Acc_{\text{W-num}}$ 等分别评估单项的预测准确率. 另外在子分类任务中也可以应用机器学习分类模型的常规评估方法如准确率、召回率、$F1$-score 等.

5 总结和展望

将自然语言转换为 SQL 查询是一个语义分析问题, 在智能问答、商业智能等领域都有实际的应用场景, 当前学术界已经对 NL2SQL 任务进行了大量的研究, 从早期的基于模板、规则和统计的模型, 以及最近广泛使用的基于深度学习的方法, NL2SQL 在英文领域已经取得了显著的进展. 中文 NL2SQL 任务的研究开展较晚, 2019 年 TableQA 和 CSpider 数据集的发布, 为中文 NL2SQL 任务的研究创造了有利条件. 近两年发布的 DuSQL 和 CHASE 更加丰富了中文数据集, 包括了从单领域、单表、简单 SQL 到多领域、多表、复杂 SQL、单轮对话、多轮对话等不同维度的数据, 场景逐步丰富, 难易程度循序渐进, 促进了该领域的发展.

中文 NL2SQL 的研究才刚刚开始, 未来可考虑从如下几个方面进行.

(1) 进一步提升模型的准确率. 目前中文 NL2SQL 任务的准确率还有较大的提升空间, 特别是 CSpider 数据集标注为“困难”和“极难”的数据. 而 CHASE 数据集由于难度较大, 尤其是涉及多轮场景的数据, 相关中文模型的研究就更少.

(2) 提升对中文语义的理解力. 在实际应用中要考虑用户的表达歧义问题. 在现实的应用场景中, 用户输入的自然语言问句难免存在表达歧义或错别字的问题, 而这些问题将会导致模型对于语义的理解出现偏差. 对语义的理解除了自然语言问句外, 对数据库的表, 也需要考虑包含的语义知识.

(3) 提升零样本的预测准确率. 当前的模型大多为有监督学习, 其训练过程基于人工标注的数据, 而标注这些数据需要较高的成本. 而在实际应用中, 存在着大量的少标注数据或无标注数据, 英文数据集上应用图神经网络模型在零样本的预测取得了一定的效果, 如何通过少量标注数据来实现中文 NL2SQL 任务, 也是

未来值得探究的一个方向.

(4) 基于大语言模型的应用. chatGPT 等大语言模型在零样本和领域泛化能力方面表现优秀, 英文 Spider 和 WikiSQL 数据集应用大语言模型后成绩提升显著. 在中文领域, 大语言模型如何应用于中文 NL2SQL 任务也将是重要的研究方向.

参考文献

1 Woods WA. Progress in natural language understanding: An application to lunar geology. Proceedings of the 1973 National Computer Conference and Exposition. New York: ACM, 1973. 441–450.

2 Sacerdoti ED. Language access to distributed data with error recovery. Proceedings of the 5th International Joint Conference on Artificial Intelligence. Cambridge: Morgan Kaufmann Publishers Inc., 1977. 196–202.

3 Warren DHD, Pereira FCN. An efficient easily adaptable system for interpreting natural language queries. Computational Linguistics, 1982, 8(3–4): 110–122.

4 Popescu AM, Etzioni O, Kautz H. Towards a theory of natural language interfaces to databases. Proceedings of the 8th International Conference on Intelligent User Interfaces. Miami: ACM, 2003. 149–157.

5 Price PJ. Evaluation of spoken language systems: The ATIS domain. Proceedings of the 1990 Workshop on Speech and Natural Language. Stroudsburg: ACL, 1990. 91–95.

6 Tang LR, Mooney RJ. Using multiple clause constructors in inductive logic programming for semantic parsing. Proceedings of the 12th European Conference on Machine Learning. Freiburg: Springer, 2001. 466–477.

7 Iyer S, Konstas I, Cheung A, *et al*. Learning a neural semantic parser from user feedback. Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). Vancouver: ACL, 2017. 963–973.

8 Li F, Jagadish HV. Constructing an interactive natural language interface for relational databases. Proceedings of the VLDB Endowment, 2014, 8(1): 73–84. [doi: 10.14778/2735461.2735468]

9 Zhong V, Xiong CM, Socher R. Seq2SQL: Generating structured queries from natural language using reinforcement learning. arXiv:1709.00103, 2017.

10 Xu XJ, Liu C, Song D. SQLNet: Generating structured queries from natural language without reinforcement learning. arXiv:1711.04436, 2017.

11 Yu T, Li ZF, Zhang ZL, *et al*. TypeSQL: Knowledge-based type-aware neural text-to-SQL generation. Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. New Orleans: ACL, 2018. 588–594.

12 Sun YB, Tang DY, Duan N, *et al*. Semantic parsing with syntax- and table-aware SQL generation. Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics. Melbourne: ACL, 2018. 361–372.

13 Dong L, Lapata M. Coarse-to-fine decoding for neural semantic parsing. Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics. Melbourne: ACL, 2018. 731–742.

14 Shi TZ, Tatwawadi K, Chakrabarti K, *et al*. IncSQL: Training incremental text-to-SQL parsers with non-deterministic oracles. arXiv:1809.05054, 2018.

15 Hwang W, Yim J, Park S, *et al*. A comprehensive exploration on WikiSQL with table-aware word contextualization. arXiv:1902.01069, 2019.

16 He PC, Mao Y, Chakrabarti K, *et al*. X-SQL: Reinforce schema representation with context. arXiv:1908.08113, 2019.

17 Lyu Q, Chakrabarti K, Hathi S, *et al*. Hybrid ranking network for text-to-SQL. arXiv:2008.04759, 2020.

18 Xu K, Wang YB, Wang YL, *et al*. SeaD: End-to-end text-to-SQL generation with schema-aware denoising. Proceedings of the 2022 Findings of the Association for Computational Linguistics. Seattle: ACL, 2022. 1845–1853.

19 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. Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing. Brussels: ACL, 2018. 3911–3921.

20 Yu T, Yasunaga M, Yang K, *et al*. SyntaxSQLNet: Syntax tree networks for complex and cross-domain text-to-SQL task. Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing. Brussels: ACL, 2018. 1653–1663.

21 Lee D. Clause-wise and recursive decoding for complex and cross-domain text-to-SQL generation. Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP). Hong Kong: ACL, 2019. 6045–6051.

22 Guo JQ, Zhan ZC, Gao Y, *et al*. Towards complex text-to-SQL in cross-domain database with intermediate representation. Proceedings of the 57th Annual Meeting of the

Association for Computational Linguistics. Florence: ACL, 2019. 4524–4535.

23 Choi DH, Shin MC, Kim EG, *et al*. RYANSQL: Recursively applying sketch-based slot fillings for complex text-to-SQL in cross-domain databases. Computational Linguistics, 2021, 47(2): 309–332.

24 Rubin O, Berant J. SmBoP: Semi-autoregressive bottom-up semantic parsing. Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. ACL, 2020. 311–324.

25 Lin XV, Socher R, Xiong CM. Bridging textual and tabular data for cross-domain text-to-SQL semantic parsing. Proceedings of the 2020 Findings of the Association for Computational Linguistics. ACL, 2020. 4870–4888.

26 Xu P, Kumar D, Yang W, *et al*. Optimizing deeper Transformers on small datasets. arXiv:2012.15355, 2021.

27 Bogin B, Gardner M, Berant J. Global reasoning over database structures for text-to-SQL parsing. Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP). Hong Kong: ACL, 2019. 3659–3664.

28 Wang BL, Shin R, Liu XD, *et al*. RAT-SQL: Relation-aware schema encoding and linking for text-to-SQL parsers. Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. ACL, 2019. 7567–7578.

29 Chen Z, Chen L, Zhao YB, *et al*. ShadowGNN: Graph projection neural network for text-to-SQL parser. Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. ACL, 2021. 5567–5577.

30 Cao RS, Chen L, Chen Z, *et al*. LGESQL: Line graph enhanced text-to-SQL model with mixed local and non-local relations. Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing. ACL, 2021. 2541–2555.

31 Hui BY, Geng RY, Wang LH, *et al*. S^2SQL: Injecting syntax to question-schema interaction graph encoder for text-to-SQL parsers. Proceedings of the 2022 Findings of the Association for Computational Linguistics. Dublin: ACL, 2022. 1254–1262.

32 Tie J, Fan ZQ, Sun C, *et al*. INSL: Text2SQL generation based on inverse normalized schema linking. Proceedings of the 4th International Conference on Artificial Intelligence in China. Springer, 2023. 195–202.

33 Yu T, Zhang R, Yasunaga M, *et al*. SParC: Cross-domain semantic parsing in context. Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics. Florence: ACL, 2019. 4511–4523.

34 Yu T, Zhang R, Er HY, *et al*. CoSQL: A conversational text-to-SQL challenge towards cross-domain natural language interfaces to databases. Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing. Hong Kong: ACL, 2019. 1962–1979.

35 Zhang R, Yu T, Er HY, *et al*. Editing-based SQL query generation for cross-domain context-dependent questions. Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP). Hong Kong: ACL, 2019. 5338–5349.

36 Kelkar A, Relan R, Bhardwaj V, *et al*. Bertrand-DR: Improving text-to-SQL using a discriminative re-ranker. arXiv:2002.00557, 2020.

37 Zheng YZ, Wang HB, Dong BH, *et al*. HIE-SQL: History information enhanced network for context-dependent text-to-SQL semantic parsing. Proceedings of the 2022 Findings of the Association for Computational Linguistics. Dublin: ACL, 2022. 2997–3007.

38 Dou LX, Gao Y, Pan MY, *et al*. UniSAr: A unified structure-aware autoregressive language model for text-to-SQL. arXiv:2203.07781, 2022.

39 Qi JX, Tang JY, He ZW, *et al*. RASAT: Integrating relational structures into pretrained Seq2Seq model for text-to-SQL. Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing. Abu Dhabi: ACL, 2022. 3215–3229.

40 Xiao DL, Chai LZ, Zhang QW, *et al*. CQR-SQL: Conversational question reformulation enhanced context-dependent text-to-SQL parsers. Proceedings of the 2022 Findings of the Association for Computational Linguistics. Abu Dhabi: ACL, 2022. 2055–2068.

41 Pourreza M, Rafiei D. DIN-SQL: Decomposed in-context learning of text-to-SQL with self-correction. arXiv: 2304.11015, 2023.

42 Sun RX, Arik SO, Nakhost H, *et al*. SQL-PaLM: Improved large language model adaptation for text-to-SQL. arXiv:2306.00739, 2023.

43 李保利, 周锡令, 胡景凡. 数据库汉语查询接口WTCDIS系统的设计与实现. 中文信息学报, 1999, 13(6):

26–33, 60.

44 孟小峰, 王珊. 数据库自然语言查询系统 Nchiql 中语义依存树向 SQL 的转换. 中文信息学报, 2001, 15(5): 40–45. [doi: 10.3969/j.issn.1003-0077.2001.05.007]

45 Shen R, Sun G, Shen H, *et al*. SPSQL: Step-by-step parsing based framework for text-to-SQL generation. arXiv: 2305.11061, 2023.

46 Min QK, Shi YF, Zhang Y. A pilot study for Chinese SQL semantic parsing. Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing. Hong Kong: ACL, 2019. 3652–3658.

47 Sun NY, Yang XF, Liu YF. TableQA: A large-scale Chinese text-to-SQL dataset for table-aware SQL generation. arXiv:2006.06434, 2020.

48 Zhang XY, Yin FJ, Ma GJ, *et al*. M-SQL: Multi-task representation learning for single-table Text2SQL generation. IEEE Access, 2020, 8: 43156–43167. [doi: 10.1109/ACCESS.2020.2977613]

49 Zhang X, Yin F, Ma G, Ge *et al*. F-SQL: Fuse table schema and table content for single-table Text2SQL generation. IEEE Access, 2020, 8: 136409–136420. [doi: 10.1109/ACCESS.2020.3011747]

50 Wang LJ, Zhang A, Wu K, *et al*. DuSQL: A large-scale and pragmatic Chinese text-to-SQL dataset. Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing. ACL, 2020. 6923–6935.

51 Guo JQ, Si ZL, Wang Y, *et al*. CHASE: A large-scale and pragmatic Chinese dataset for cross-database context-dependent text-to-SQL. Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing. ACL, 2021. 2316–2331.

(校对责编: 孙君艳)