Operations Research and Fuzziology 运筹与模糊学, 2024, 14(1), 83-90
Published Online February 2024 in Hans. https://www.hanspub.org/journal/orf
https://doi.org/10.12677/orf.2024.141008

Hans汉斯

基于问句语义图神经网络的中文问句生成SQL语句研究

张海芳，何清龙

贵州大学数学与统计学院，贵州 贵阳

收稿日期：2023年11月8日；录用日期：2023年11月28日；发布日期：2024年2月18日

摘　要

自然语言问句转为结构化查询语句(Text-to-SQL)是语义解析领域中热点研究之一，其目标是将自然语言问句转化为数据库可以理解且执行的结构化查询语句。现有研究大部分仅考虑数据库层面的关联信息，忽略了问句中的实体关系信息的重要性。为了提高模型捕捉问句中语义的有用信息，本文在IGSQL模型基础上，引入问句中实体之间的图网络信息，通过注意力机制来自动学习问句和数据库模式之间的关联。在Chase数据集上的实验结果表明，本文提出模型的完全匹配率达到46.2%。相比较于基线模型，完全匹配率提升了6.3%。

关键词

Text-to-SQL，自然语言处理，图神经网络，中文多表SQL语句生成

Research on Chinese Question Generation SQL Statement Based on Question Semantic Graph Neural Network

Haifang Zhang, Qinglong He

School of Mathematics and Statistics, Guizhou University, Guiyang Guizhou

Received: Nov. 8th, 2023; accepted: Nov. 28th, 2023; published: Feb. 18th, 2024

Abstract

The conversion of natural language questions into structured query statements (Text to SQL) is one of the hot research topics in the field of semantic parsing, with the goal of transforming natural language questions into structured query statements that can be understood and executed by databas-

文章引用: 张海芳, 何清龙. 基于问句语义图神经网络的中文问句生成 SQL 语句研究[J]. 运筹与模糊学, 2024, 14(1): 83-90. DOI: 10.12677/orf.2024.141008

es. Most existing research only considers relational information at the database level, ignoring the importance of entity relationship information in questions. In order to improve the model's ability to capture useful semantic information in questions, this paper introduces graph network information between entities in questions based on the IGSQL model, and automatically learns the association between questions and database patterns through attention mechanisms. The experimental results on the Chase dataset show that the proposed model has an exact matching accuracy of 46.2%. Compared to the baseline model, the exact matching accuracy has increased by 6.3%.

Keywords

Text-to-SQL, Natural Language Processing, Graph Neural Network, Chinese Multi-Table SQL Statements Generation

1. 引言

进入 21 世纪以来，在计算机的高速发展下，人类已经从工业革命进入信息时代。现实生活中各行各业每天都会产生大量的数据，而这些数据中隐藏的大量信息可以为日常生活带来巨大的便利，如何有效地存储并利用这些数据成为重中之重。数据库技术的发展可以很好地解决这一问题，它将数据按照一定的规则进行存储，同时也会提供对于数据的查、改、删、增等一系列操作，而实现这些操作，往往都会用到结构化查询语言(Structure Query Language, SQL)。为了帮助非专业的用户从数据库中查询到所需要的信息，提出了基于深度学习的自然语言–数据库接口(Natural Language Interface to Database, NLIDB)，极大地降低了用户与数据库之间的交互门槛[1]。

NLIDB 的核心任务是根据已有数据库信息，将自然语言问句映射到特定功能的 SQL 语句(Text-to-SQL)。Text-to-SQL 是自然语言处理领域里语义解析的一个子任务。语义解析旨在将自然语言表达转换为结构化的语义表示，便于计算机能够理解和处理，其涉及语言学、计算语言学以及认知语言等多个学科，近几年获得了广泛的关注[2] [3] [4]。Text-to-SQL 的深度学习模型需结合自然语言问句和数据库本身的信息，提取出关联的数据库表名、列名以及问句的意图模式。然而数据库本身复杂的内部结构以及中文自然语言问句表达方式多变等因素会给深度学习模型带来巨大的挑战，主要挑战有：1) 数据库中每张表存在表名、列名以及列名所对应的值，除此之外，还有主键、外键等关联信息，如何学习从这些复杂信息到结构化语句的映射是非常困难的；2) 相比较于英文的自然语言问句，中文问句的语法结构较为灵活，动词和宾语的位置可以交换，给模型的学习带来了极大的挑战[5]。图神经网络因其能够捕捉节点之间的局部结构信息和全局拓扑特征，近年来，被越来越多的深度学习网络引入。图神经网络(Graph Neural Network, GNN) [6]通过将原始数据转为关系图谱作为输入，采用聚合节点的邻居信息，在图结构上进行线性变换、卷积等操作，能够准确表示各元素之间的关联信息。

本文的主要贡献如下：

(1) 本文借助图神经网络，引入自然语言问句中的语义依存信息，丰富了问句中每个单词的表示，从而提升识别关键词的准确率。

(2) 基于中文 Text-to-SQL 任务数据集 Chase 上进行验证，实验表明，本文所提出的方法可以有效提升 SQL 的匹配准确率。

2. 相关工作

目前，基于深度神经网络的 Text-to-SQL 模型通常由编码和解码模块构成，编码模块主要目的是提取问句与数据库模式的特征信息，便于解码模块准确生成 SQL 语句；解码模块利用编码模块所提取的特征信息进行 SQL 语句生成。

SQLNet [7]通过列名注意力机制来增强模型捕捉问题与数据库模式之间的联系，然后在预先定义的 SQL 模板上使用指针网络来补全 SQL 信息。随着图神经网络的发展，结构化的数据库信息逐渐使用图神经网络进行编码。GNN-SQL [8]使用图神经网络对数据库中表的结构形成全局表示，以此让模型在未见过的数据库模式上具有更好的泛化能力。Global-GNN [9]通过引入全局节点，强化了问句和模式的全局化特征。然而上述方法仅仅考虑数据库层面的关联信息，没有考虑到问句中单词和单词之间的关联。

Zhong 等首次在大规模 SQL 生成数据集 WikiSQL 上使用序列生成的方法，提出了 Seq2SQL [10]模型。Seq2SQL 模型的主要创新是在 Seq2Seq 模型的基础上添加了输出模板，并将完整的 SQL 序列拆解成 3 个子任务进行学习，即 agg 的生成、where 子句的生成和 select 的生成，在此基础上引入了强化学习方式。SQLNet [7]在 Seq2SQL 基础上对子任务设计上做了进一步的细化，将 where 子句拆解为 4 个子任务，并且模型还引入了列注意力机制以强调自然语言问句中涉及列名的信息。基于树形结构的 Seq2Tree [11]采用了层级式树解码方法，可以很好地捕捉 SQL 语句的组成结构。SyntaxSQLNet [12]利用堆栈来控制解码的进程，便于实现嵌套查询这类复杂句式的生成。

3. 问题描述与模型框架

3.1. 问题描述

Text-to-SQL 任务的输入是一个查询语句以及其相应的数据库模式，输出是 SQL 语句。具体而言，模型需要根据所给定的长度为 L 个字符的问句 $Q=\left[q_1,q_2,q_3,\cdots,q_L\right]$ 和相应的数据库模式 S 生成结构化查询语句 Y。数据库模式指表名、列名、列类型和主外键等数据表逻辑结构信息，这里的数据库模式 $S=\left[tab_1.col_1,tab_1.col_2,tab_1.col_3,\cdots,tab_n.col_k\right]$，其中 tab 指的是表名，col 指的是列名，n 为数据库包含表的数量，k 为数据库中第 n 个表所包含的列名数。

3.2. 模型框架

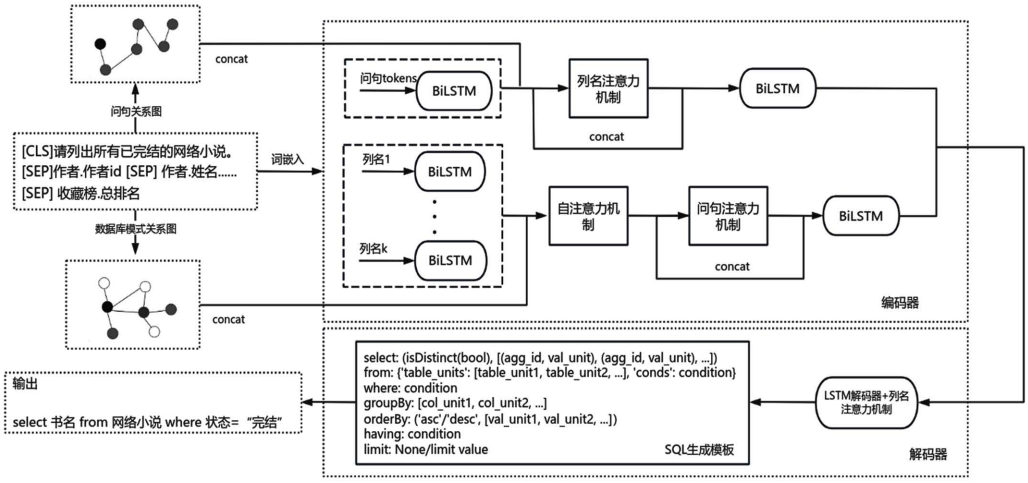

Figure 1. The overall framework of the model in this article
图 1. 本文模型的整体框架

本文提出的基于问句语义图神经网络的 Text-to-SQL 模型采用序列到序列的框架，如图 1 所示，主要由 3 个模块组成：1) 图神经网络部分，包含数据库中各个表之间的关系、各个表中列名之间的关系及自然语言问句中语义依存关系；2) 编码器部分，通过图神经网络将这些关系融入词嵌入向量中，再使用注意力机制将问句、数据库模式联合编码为隐藏向量表示；3) 解码器部分，使用带有注意力机制([13], pp. 199-205)的长短期记忆网络(Long Short-Term Memory, LSTM) ([13], pp. 145-149)生成 SQL 语句。

4. 模型设计

4.1. 图网络构建

现有 Text-to-SQL 模型根据数据库信息构建图网络，仅仅包含表名－表名、表名－列名以及列名－列名 3 种关系类型[8] [9]。本文在此基础上，还添加了自然语言问句中各个词之间的语义依存关系，有施事关系、当事关系、受事关系等 55 种关系类型[14]，进一步让模型从现有的数据中学习到有用的信息。

本文根据数据库和问句分别构建了各自对应的图网络，其中，数据库图网络的节点为表名和列名，问句图网络的节点为哈工大 LTP 分词[14]的结果。构建的图网络所包含的关系类型具体有：

(1) 表名→列名：表的主键是列名；表中存在列名；表中不存在列名。

(2) 表名→表名：两个表通过外键连接；两个表互为外键连接；两个表不存在直接连接。

(3) 列名→表名：列名是表的主键；列名属于表的其中一列；列名不属于表的一列。

(4) 列名→列名：两个列名属于同一张表；两个列名不属于同一张表；一个列名是另一个列名的外键。

(5) 问句词→问句词：55 种语义依存关系中任意一种。

4.2. 编码器

为解决未登录词问题，本文首先使用 RoBERTa 预训练语言模型[15]获得每个字符的向量，在此基础上根据分词结果，对词汇中包含字符的向量加和取平均获得词嵌入表示 q^{init}， c^{init}，其中 q^{init} 是自然语言问句的词向量表示，c^{init} 是数据库中列名的词向量表示，然后对问句和数据库模式构建图网络 G_q 和 G_c，这里以问句图网络为例，获得最终词向量的具体过程如下：

(1) 问句图网络的节点是 LTP 的分词结果，根据 LTP 的语义依存分析结果，构建邻接矩阵 R_g，矩阵中元素 r_{ij} 表示第 i 行，第 j 列所对应的关系。

(2) 将问句的词向量 q^{init} 通过全连接层和激活函数得到新的词向量 q_{new}：

$$q_{new} = Leaky\,\mathrm{Re}\,LU\left(linear\left(q^{init}\right)\right) \tag{1}$$

(3) 对 q_{new} 做步骤(2)的变换，得到矩阵 w，再将其与 q_{new} 的转置矩阵 q_{new}^t 相乘得到 $output_1$：

$$w = Leaky\,\mathrm{Re}\,LU\left(linear\left(q_{new}\right)\right) \tag{2}$$

$$output_1 = w \cdot q_{new}^t / 10 \tag{3}$$

(4) 根据关系矩阵 R_g 中元素是否为 0 得到矩阵 M，基于 M 中元素值，生成新的 $output_2$。

(5) $output_2$ 经激活函数与 q_{new} 相乘得 $output_3$，再将 $output_3$ 做步骤(2)的变换与 q^{init} 相加得最终编码向量 $output$：

$$output_3 = soft\max\left(output_2\right) \cdot q_{new} \tag{4}$$

$$output = Leaky\,\mathrm{Re}\,LU\left(linear\left(output_3\right)\right) + q^{init} \tag{5}$$

4.3. 解码器

解码器使用 IGSQL 模型中基于注意力机制的 LSTM 网络解码 SQL 语句。首先将问句词向量和模式词向量与 LSTM 的隐藏层向量做注意力操作，然后根据这一结果计算 SQL 关键词和数据库模式项这两类的重要系数，最后在重要系数基础上，选择对应的解码器，生成 SQL 语句中的字符。

重要系数的计算方式如下：

$$P\left(word_i\right)=\sigma\left(W_i\cdot tanh\left(W_o\left[o;c\right]+b_o\right)+b_i\right),i\in\left\{SQL\_reserved,schema\right\} \tag{6}$$

其中，o 是 LSTM 解码器的隐藏向量，c 是问句和模式词向量与 LSTM 隐藏层做注意力操作的拼接向量，$\sigma(\cdot)$ 是 sigmoid 函数。

5. 实验与结果分析

5.1. 实验环境及数据集

本文的实验是在操作系统 Ubuntu20.04 下进行的，GPU 为 NVIDIA RTX A4000 16GB，开发编程语言为 Python3.9，深度学习模型框架为 PyTorch [16]。

CHASE 是由微软亚洲研究院和北航、西安交大联合提出的首个大规模中文跨领域上下文依赖的 Text2SQL 数据集，其由 CHASE-C 和 CHASE-T 两部分组成，分别来自 DuSQL 数据集、SParC 数据集[17]。它包含了 5459 个对话以及相应的 SQL 查询语句，这些对话覆盖了 280 个数据库。为方便研究人员进行研究，CHASE 每个问题都有丰富的语义注释，并且划分出训练集、测试集以及验证集部分。但考虑到本文研究的是上下文无关的查询问句生成 SQL，因此，选取每轮对话中的第一个问题以及对应的 SQL 语句构成本文的数据集。

5.2. 评价指标

本文使用两个指标评价模型在 CHASE 数据集上的效果，分别为部分匹配率 (Partial Matching Accuracy, PM)和完全匹配率(Exact Matching Accuracy, EM)，前者是将 SQL 拆解为“SELECT”，“WHERE”等几个部分，然后计算每个部分真实值与预测值的 F1 值；后者是在无顺序要求的情况下，衡量 SQL 语句预测值与真实值之间的匹配程度[18]。

5.3. 实验设置

本文使用中文 RoBERTa 作为预训练模型，在学习率为 1e-5 下进行微调，其可以处理的最大长度为 512，但本文的训练样本最大长度不超过 300，因此将 300 作为 RoBERTa 的输入长度。训练时迭代次数 epoch 为 10，批量大小 batch size 为 32，采用的优化器是 Adam，使用 warmup 机制训练，Dropout 层损失信息比例设置为 0.5，模型学习率为 0.003。

5.4. 对比实验分析

本文使用 EditSQL [19]和 IGSQL [20]模型作为基线模型进行对比，为了下文叙述的方便，本文提出的模型记为 Ours。表 1 是不同模型在 Chase 测试集上的实验结果。

Table 1. Experimental results of different models
表 1. 不同模型的实验结果

模型	EM
EditSQL	40.5

续表

IGSQL	39.9
Ours	46.2

从表 1 的实验结果可以知道，模型引入问句中语义依存关系的信息可以提高模型的匹配率。相比较原有模型，Ours 在完全匹配率上提升了 6.3%，具有显著效果。

Table 2. Analysis of experimental results of IGSQL and Ours model
表 2. IGSQL 与 Ours 模型的实验结果分析

例子 1	问句	我国哪家医院的重点专科是最多的？
	EditSQL	select 医院名 from 医院 **where** 重点专科数量 = (select max (重点专科数量) from 医院)
	IGSQL	select 医院名 from 医院 **order by** 重点专科数量 desc limit 1
	Ours	select 医院名 from 医院 **where** 重点专科数量 = (select max (重点专科数量) from 医院)
	解释	IGSQL 将“最多”理解成对重点专科数量进行排序。
例子 2	问句	什么时候过端午节呢？
	EditSQL	select **节日** from 民俗节日 where **时间** = “端午节”
	IGSQL	select **节日** from 民俗节日 where 节日 = “端午节”
	Ours	select **时间** from 民俗节日 where 节日 = “端午节”
	解释	EditSQL 对时间和节日的理解不充分；IGSQL 没有理解问句想要获取的是时间，而不是节日。
例子 3	问句	有哪些输入法软件？
	EditSQL	select 名称 from 软件 where **名称** = “输入法”
	IGSQL	select 名称 from 软件 where **名称** = “输入法”
	Ours	select 名称 from 软件 where **用途** = “输入法”
	解释	EditSQL 和 IGSQL 对这句话的理解很表面，没有获取到“哪些”指的是数据库模式中的“用途”。

表 2 是 EditSQL 和 IGSQL 与 Ours 模型的实验结果例子分析，其中 EditSQL、IGSQL 生成的是错误的 SQL，Ours 生成的是正确的 SQL 语句。表 2 中的解释是对 EditSQL 和 IGSQL 生成错误 SQL 的原因分析，可见 Ours 可以更好的将问句中关键信息与数据库模式对齐。

为了更细致地分析本文提出的模型在不同 SQL 子句上的性能表现，表 3 给出了模型在各个子句上的 F1 值。从表 3 可以看出模型随着难度的增加，各子句的 F1 值并没有一直下降，其中 select 子句在困难程度下，F1 值达 80.3%，where 子句中，F1 值最高的是特别困难程度下。总的来说，对于生成复杂 SQL 语句来说，模型具有良好效果。

Table 3. F1 values of SQL clauses under different levels of difficulty
表 3. 不同难度下 SQL 各子句 F1 值

子句类别	简单	中等	困难	特别困难
select	90.5	57.0	80.3	55.0

续表

where	64.9	54.2	48.1	68.8
group	100.0	100.0	100.0	66.7
order	100.0	60.0	43.9	52.6

5.5. 消融实验分析

为了分析不同模块对 Ours 模型带来的效果增益，本节对 Ours 模型进行消融来验证中文 RoBERTa 词向量模块和问句的语义依存关系图模块对模型性能的影响。消融实验结果如表 4 所示。

Table 4. Ablation experiment results of the Ours
表 4. Ours 消融实验结果

模型	EM
Ours	46.2
移除 RoBERTa 词向量模块	41.7
移除语义依存关系图模块	40.5
两者都移除	39.9

从表 4 的实验结果可以看出，完整模型相比不同的消融模型取得的效果最佳。对比移除 RoBERTa 词向量模块，移除语义依存关系图对模型的影响较大。因此，本文引入问句的语义依存关系图对于模型性能的提升起到一定作用。

6. 结束语

针对语义解析领域的 Text-to-SQL 任务，本文提出一种基于问句语义图神经网络的模型。根据自然语言问句与数据库模式信息分别构建关系图谱，通过图邻接矩阵的方式，将关系信息编码到词向量中。在 Chase 数据集上的实验结果表明，相比于 EditSQL 模型和 IGSQL 模型，本文所提出的模型能够利用语义依存信息更好捕捉问句中语义的信息，有效提升了模型在多表查询中的匹配准确率。然而，在现实的应用中，因自然语言交互的随意性，SQL 生成任务仍存在诸多困难，比如零次学习(zero-shot)等。针对上述挑战，未来的研究可以通过改述自然语言问句来增强样本容量，或者引入外部知识(比如知识图谱等)来增强模型的表示能力。

参考文献

[1] 潘璇, 徐思涵, 蔡祥睿, 等. 基于深度学习的数据库自然语言接口综述[J]. 计算机研究与发展, 2021, 58(9): 1925-1950.

[2] Yang, J., Jiang, H., Yin, Q., *et al*. (2022) Seqzero: Few-Shot Compositional Semantic Parsing with Sequential Prompts and Zero-Shot Models. *Findings of the Association for Computational Linguistics*, Seattle, July 2022, 49-60. https://doi.org/10.18653/v1/2022.findings-naacl.5

[3] Iyer, S., Cheung, A. and Zettlemoyer, L. (2019) Learning Programmatic Idioms for Scalable Semantic Parsing. *Proceedings of the* 2019 *Conference on Empirical Methods in Natural Language Processing and the* 9*th International Joint Conference on Natural Language Processing*, Hong Kong, November 2019, 5426-5435. https://doi.org/10.18653/v1/D19-1545

[4] Li, D. and Lapata, M. (2016) Language to Logical Form with Neural Attention. *Proceedings of the* 54*th Annual Meeting of the Association for Computational Linguistics*, Volume 1, 33-43.

[5] 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[J]. 计算机系统应用, 2023, 32(12): 32-42.

[6] Wu, Z., Pan, S., Chen, F., *et al*. (2021) A Comprehensive Survey on Graph Neural Networks. *IEEE Transactions on Neural Networks and Learning Systems*, **32**, 4-24. https://doi.org/10.1109/TNNLS.2020.2978386

[7] Xu, X., Liu, C. and Song, D. (2017) SQLNet: Generating Structured Queries from Natural Language without Rein-

forcement Learning.

[8] Bogin, B., Gardner, M. and Berant, J. (2019) Representing Schema Structure with Graph Neural Networks for Text-to-SQL Parsing. *Proceedings of the* 57*th Annual Meeting of the Association for Computational Linguistics*, Florence, July 2019, 4560-4565. https://doi.org/10.18653/v1/P19-1448

[9] Bogin, B., Gardner, M. and Berant, J. (2019) Global Reasoning over Database Structures for Text-to-SQL Parsing. *Proceedings of the* 2019 *Conference on Empirical Methods in Natural Language Processing and the* 9*th International Joint Conference on Natural Language Processing* (*EMNLP-IJCNLP*), Hong Kong, November 2019, 3659-3664. https://doi.org/10.18653/v1/D19-1378

[10] Zhong, V., Xiong, C. and Socher, R. (2017) Seq2SQL: Generating Structured Queries from Natural Language Using Reinforcement Learning.

[11] Dong, L. and Lapata, M. (2016) Language to Logical Form with Neural Attention. *Proceedings of the* 54*th Annual Meeting of the Association for Computational Linguistics*, Volume 1, 33-43. https://doi.org/10.18653/v1/P16-1004

[12] Yu, T., Yasunaga, M., Yang, K., *et al*. (2018) 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, October-November 2018, 1653-1663. https://doi.org/10.18653/v1/D18-1193

[13] 邱锡鹏. 神经网络与深度学习[M]. 北京: 机械工业出版社, 2020.

[14] Che, W., Feng, Y., Qin, L., *et al*. (2021) N-LTP: An Open-Source Neural Language Technology Platform for Chinese. *Proceedings of the* 2021 *Conference on Empirical Methods in Natural Language Processing*: *System Demonstrations*, November 2021, 42-49. https://doi.org/10.18653/v1/2021.emnlp-demo.6

[15] Liu, Y., Ott, M., Goyal, N., *et al*. (2019) RoBERTa: A Robustly Optimized BERT Pretraining Approach.

[16] Paszke, A., Gross, S., Massa, F., *et al*. (2019) PyTorch: An Imperative Style, High-Performance Deep Learning Library. 33*rd Annual Conference on Neural Information Processing Systems* (*NeurIPS* 2019), Vancouver, 8-14 December 2019, 8026-8037.

[17] Guo, J.Q., Si, Z.L., Wang, Y., *et al*. (2021) Chase: A Large-Scale and Pragmatic Chinese Dataset for Cross-Database Context-Dependent Text-to-SQL. *Proceedings of the* 59*th Annual Meeting of the Association for Computational Linguistics and the* 11*th International Joint Conference on Natural Language Processing*, Volume 1, 2316-2331. https://doi.org/10.18653/v1/2021.acl-long.180

[18] 赵志超, 游进国, 何培蕾, 李晓武. 数据库中文查询对偶学习式生成 SQL 语句研究[J]. 中文信息学报, 2023, 37(3): 164-172.

[19] Zhang, R., Yu, T., *et al*. (2019) Editing-Based SQL Query Generation for Cross-Domain Context-Dependent Questions. *Proceedings of the* 2019 *Conference on EMNLP and the* 9*th IJCNLP*, Hong Kong, November 2019, 5338-5349. https://doi.org/10.18653/v1/D19-1537

[20] Cai, Y.T. and Wan, X.J. (2020) IGSQL: Database Schema Interaction Graph Based Neural Model for Context-Dependent Text-to-SQL Generation. *Proceedings of the* 2020 *Conference on EMNLP*, November 2020, 6903-6912. https://doi.org/10.18653/v1/2020.emnlp-main.560