Text-to-SQL 领域调研报告 III

日期：2023 年 3 月 12 日

摘　要

本次的报告基于上次收到老师的邮件指点，并提出的 4 个需要思考的问题出发。从鲁棒性与泛化性、多轮问题提升方向、单轮问题 sota 模型改进思路以及评价指标四个方面入手，经过对鲁棒性与泛化性方面的研究论文以及回顾先前调研过的论文，串起了近些年这方面的研究历史，并发现了多轮任务向单轮任务转变的契机。在单轮任务方面，我从 Graphix-T5 的模型框架上找到思路，从多模态的角度出发，提出了针对该模型的改进思路，并在最后对老师上一封邮件中提到的问题做了总结回答。

关键词: Text-to-SQL，鲁棒性与泛化性，多轮问题向单轮问题转化，单轮 sota 模型改进

1　工作汇报

这段时间的调研我首先集中于 Text-to-SQL 领域鲁棒性方面的研究，在过程中对该领域面临的挑战有了更深的理解，并发现了该领域另一个挑战——泛化性。借此思路我又查阅了先前调研时单轮和多轮问题的一些论文中模型在鲁棒性和泛化性方面的阐述，发现了两个领域在此方面的研究并不同步，多轮问题在此方面涉及较少，单轮问题则是提出了能反应模型鲁棒性和泛化性的验证集，并已经在近期的优秀模型中使用。此外，在对多轮问题的难点挑战调研中，发现使用对话式问题再造 (CQR) 将多轮问题转化为单轮问题的技术。在最后思考对单轮任务 sota 模型 Graphix-T5 的改进时联想到先前调研过的多模态领域知识，并以此作为改进模型框架的思路。

2　鲁棒性与泛化性

鲁棒性: 在 Text-to-SQL 领域的鲁棒性在多篇文章 [**1**, **2**, **3**, **7**] 中被定义为在自然语言问题上，或者数据库表名列名上修改、替换部分词语或语言风格使得问题更贴近实际使用。

泛化性: 在 Text-to-SQL 领域的泛化性在多篇文章 [**3**, **4**, **5**, **6**] 中被定义为应用于不同领域，不同数据库。模型能否应对全新的问题，并生成有效的 SQL 语句。

近期关于 Text-to-SQL 领域鲁棒性和泛化性的研究，先是在 2020 到 2021 年被提出的研究鲁棒性方面的 Spider syn[**1**] 和 Spider realistic[**2**]，与研究泛化性方面的 Spider ssp[**5**] 和 Spider dk[**4**]，随后 2022 年出现了提出解决鲁棒性与泛化性的 TKK[**3**] 框架与提出使用对抗方式来提升模型鲁棒性 [**7**]。最后到今年先后出现了针对鲁棒性的最新研究 DR.SPIDER[**8**] 与针对多轮 Text-to-SQL 领域的泛化性的研究 [**6**]。而在近期 Text-to-SQL 领域对模型的研究如 Graphix-T5[**9**]、RESDSQL[**10**] 等则使用先前提出的 Spider 数据集的四个变体 (Spider dk、Spider ssp、Spider syn、Spider realistic) 作为验证集，通过在这些数据集上的准确率来评判模型鲁棒性和泛化性。下面将具体阐述这段话提到的部分论文。

2.1 Spider 数据集的变体

Spider ssp: Spider ssp 是文章 Compositional Generalization and Natural Language Variation:Can a Semantic Parsing Approach Handle Both?[**5**] 对于在探究其提出的 NQG-T5 应对 Text-to-SQL 领域内组合泛化问题时，剔除的数据集，该数据集由从 Spider 得到的三组拆分数据 Spider-Length、Spider-TMCD 和 Spider-Template 组成。近期的模型通常使用在这三部分上分别的准确率来验证模型组合泛化性。

Spider syn: Spider syn 是文章 Towards Robustness of Text-to-SQL Models against Synonym Substitution[**1**] 对于 text2sql 领域模型鲁棒性问题提出的基于 Spider 基准的人工管理数据集。文章考虑到现有的 text2sql 模型通常依赖于自然语言 (NL) 问题中的单词和表模式中的标记之间的词汇匹配，从同义词入手，使用生活中常用词汇替换与数据库表、列名相关的词。

Spider realistic: Spider realistic 是文章 Structure-Grounded Pretraining for Text-to-SQL[**2**] 对于 text2sql 领域模型鲁棒性问题提出的基于 Spider 基准的人工管理数据集。提出该数据集主要目的是检测文章中提出的预训练框架在鲁棒性方面的表现。该数据集类似 Spider syn，但是替换内容不同 (例如把从句替换成更符合实际使用时人们输入的表达，以此降低与 schema 中的列名表名的直接相似性)。

Spider dk: Spider dk 是文章 Exploring Underexplored Limitations of Cross-Domain Text-to-SQL Generalization[**4**] 对于 text2sql 领域模型泛化性问题提出的基于 Spider 基准的人工管理数据集。该数据集重点关注对模型理解领域知识能力的评估。构建 Spider dk 的目的是模拟用户话语查询中涉及特定领域知识的场景。例如 “Q:How many students got accepted after the tryout? 这个的 SQL 语句中应该使用的判断是 from tryout where decision=”yes”，需要模型有基础知识背景以了解 accepted 的相关领域知识”。

2.2 鲁棒性泛化性的论文研究

TKK 框架: 文章认为解决鲁棒性一个重点是实现语义解析而不是匹配[1]；泛化可以从 zero-shot generalization 和 compositional generation[2]两方面研究，前者在当前大规模数据集 Spider、SParC、CoSQL 的训练下已被基本解决。为了解决上述两方面问题，文章提出了由任务分解、知识获取到知识合成的多阶段学习，增强模型获取一般 SQL 知识的能力。

对抗表扰动: 文章提出一种新的攻击范式来衡量 Text-to-SQL 模型鲁棒性 ADVETA(第一个鲁棒性评估指标)，并构建了一个系统的对抗训练示例生成框架，用于更好地将表格数据上下文化，极大增强了模型对 NL 侧扰动[3]的抵抗力。除了这部分的扰动，文章还讨论了关于表的扰动，即修改数据库表名、列名以及用原有数据生成有意义的列添加进数据库等的扰动，但文章中的框架在该部分扰动的改进上不如对 NL 侧扰动的改进。

Dr.Spider: 本文提出了一个基于 Spider 的综合鲁棒性指标来诊断模型鲁棒性。与 Spider syn 类似都考虑到模型从自然语言问题中寻找与 schema 匹配的单词而不是理解自然语言问题的问题。文章使用的合成数据方法从 DB、NLQ、SQL 三个方向添加扰动，其中包括上述的 Spider syn 和 Spider realistic，但是对于 Spider dk 方面的领域知识并未说明。本文对于评估模型鲁棒性提出了三个指标，而当前很多论文如 Graphix-T5、RESDSQL 等都是单纯使用的 Spider 的三个变体数据进行的鲁棒性评估。该文在今年 1 月 28 日提交，是单轮 Text-to-SQL 领域关于鲁棒性最新的研究。

[1]此观点与 Spider syn 和 Spider realistic 数据集提出的思想吻合

[2]解决重新组合在训练集中出现过的信息形成新的问题的能力

[3]指在自然语言问题上做同义词替换等修改扰动，该部分类似于 Dr.Spider 中的 NLQ 部分扰动

3 对多轮问题的思考

多轮任务带来的新难点和挑战在于前后轮次之间的上下文关联，用户可能因为上一个问题而在第二个问题中使用代词省略或者突然改变主题等。如何使用前面轮次的用户问题以及输出 SQL 就成了一个巨大挑战。为了解决上下文依赖，模型不仅要理解共引用和省略，还要防止用户焦点变化时不相关的信息集成。先前调研的多轮问题 sota 模型 MIGA[**11**] 采取将先前自然语言问题合并入后续的输入，在这次的调研中我重新查阅了论文 CQR-SQL: Conversational Question Reformulation Enhanced Context-Dependent Text-to-SQL Parsers[**12**]，发现可以使用 CQR(对话式问题重构) 从先前的对话中获取信息，将后续含有引用、省略的句子重构为完整且单独的问题，以此实现多轮 Text-to-SQL 任务向单轮 Text-to-SQL 问题转化，结合单轮 Text-to-SQL 领域 sota 模型有望较大提升多轮 Text-to-SQL 问题的准确率。

除了使用 CQR 的方法，我的另一个思路是使用对话生成系统领域的一些经验方法。因为多轮 Text-to-SQL 问题本身就是对话生成系统的一种，但这个领域我调研较少，并未形成完整体系，在此不做阐述。

4 单轮问题 sota 的改进

当前单轮问题 sota 模型为 Graphix-T5，模型结构使用图注意机制和 T5 模型。虽然在 Spider 数据集上评分最优，但在使用 Spider dk、Spider syn 等变体数据集验证时，鲁棒性和泛化性低于提出以解耦模式链接和骨架解析的模型 RESDSQL[**10**]。当前单轮问题领域模型改进主要为在 Spider 数据集上准确率的提升与在模型鲁棒性、泛化性的提升。

除了使用数据增强 (合成数据) 的方法可以较好得提高模型的鲁棒性与泛化性外，还能从模型自身结构入手。Graphix-T5 模型在 Encoding 阶段在每一层对数据库 schema 和 nlq 分别提取特征，然后加和，然后再传递进入下一层继续重复。从多模态领域的角度，该结构在数据库 schema 和自然语言两模态之间的模态融合 (fusion) 方面欠佳。考虑可以使用如图 1 所示多模态领域经典的双塔结构作为 Encoding 部分。

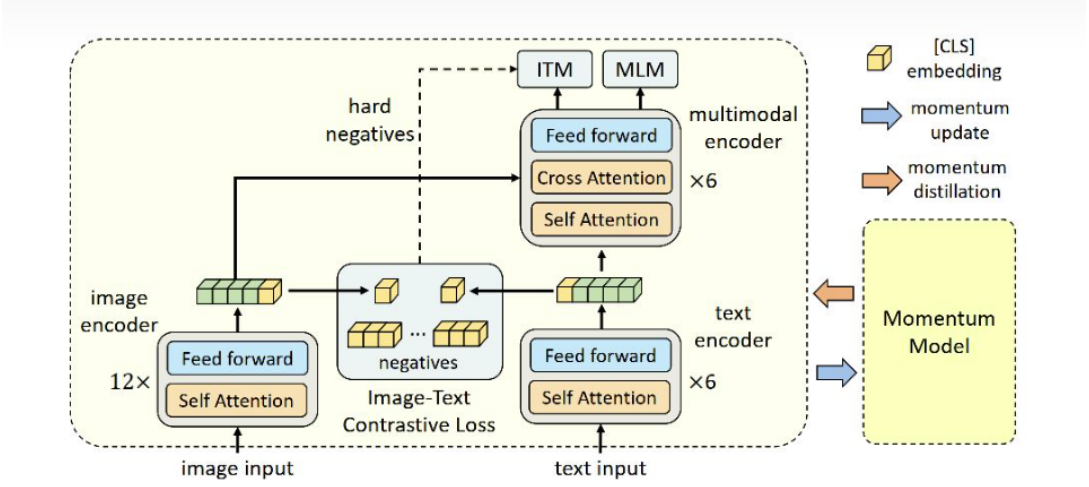

图 1: ALBEF[**13**] 模型结构

具体来说对于自然语言问题方面，使用如 Bert[**14**]、RoBERTa[**15**] 等模型作为这部分的 encoder，使用 Graphix-T5 论文中的图注意机制或者表格预训练模型 TaBERT[**16**] 等作为数据库 schema 部分的 encoder，后续对其 (align)，拼接 (concat) 在使用注意力网络进行模态融合 (fusion) 实现整体的 encoding 部分，后续 decoding 部分可以仍然使用先前的 T5[**17**] 中的 decoder 或者 GPT[**18**] 等预训练模型，最后针对输出加上 PICARD[**19**] 框架规范。

5 领域内评价指标

5.1 单轮问题

单轮问题评价指标有 Exact Match(EM) 和 Execution Accuracy(EX)，两指标在论文 Spider: A Large-Scale Human-Labeled Dataset for Complex and Cross-Domain Semantic Parsing and Text-to-SQL Task[**20**] 后被普遍用于 Text-to-SQL 领域的模型评价。Exact Match 用于评估生成的 SQL 与标准 SQL 答案的相似性，Execution Accuracy 用于反映生成的 SQL 的执行结果是否正确。前者会因为 SQL 语句不同风格，不同写法出现假阴性误判，后者则会因为偶然出现相同的答案而出现假阳性误判，但是 EX 指标出现假阳性误判概率明显小于 EM 的假阴性误判。一个有效的例子就是 RESDSQL 模型 [**10**]，其因为固定生成 SQL 语句骨架后再生成完整 SQL 语句，且在 SQL 骨架生成时统一生成偏模板化的风格，甚至将训练数据集中 SQL 语句风格进行模板化统一，所以该模型生成的 SQL 语句在 EM 指标表现较差，但 EX 指标接近 sota 模型，且鲁棒性、泛化性被验证优于 sota 模型。

5.2 多轮问题

多轮问题评价指标常用的有 Question Match(QM) 和 Interaction Match(IM), 两指标在论文 SParC: Cross-Domain Semantic Parsing in Context[**21**] 提出后被普遍应用于多轮问题模型的评估。Question Match 是所有问题的精确匹配得分，Interaction Match 是所有交互中的精确匹配得分，只有当所有预测的 SQL 子句都正确时，每个问题的精确匹配集得分 (QM) 才为 1，只有当交互中的每个问题都有精确匹配集时，每个交互的精确匹配集得分 (IM) 才为 1。两指标都针对生成的回答与标准 SQL 回答之间的相似性，都有误判假阴的可能性。

6 结论

Q1： *单轮问题的深入分析：目前单轮的 sota 是 79% 左右，这种模型从性能上讲是否还有进一步提升的空间？还存在的问题是什么？你调研中提到的鲁棒性问题，是单轮目前的最主要问题吗？数据增强（使用合成数据）的方法是解决鲁棒性问题的有效手段吗？*

A1： 首先单看 Graphix-T5 模型，我认为其本身在 Encoding 部分关于图注意机制和 T5 的融合不够充分，模型鲁棒性与泛化性不及准确率较低的 RESDSQL 模型。在调研中，单轮问题中模型主要关注准确率 (EM、EX 评分)，鲁棒性以及泛化性。从工业应用方面鲁棒性和准确性是单轮问题目前最主要的问题，因为工业应用多针对一个专门的数据库，领域较为固定。从学术研究方面三者都是单轮问题领域主要的问题，只是目前看来鲁棒性方面或许是个很好的研究切入点。在解决鲁棒性问题方面，一是可以使用有针对性 (针对同义词、生活中常用表达等) 的合成数据训练模型，二是使用新的模型架构，或者在原有模型上改进。前者在工业应用上有较大优势，但在学术研究上或许创新性欠佳。在 A4 中有对 Graphix-T5 模型可能的具体改进思路。

Q2： *多轮问题的难点与挑战：与单轮相比，多轮即可以看做是多个单轮的组合也可以看做是一个全新的问题。那跟单轮的研究相比，多轮带来的新的难点和挑战在什么地方呢（比如多轮之间的关联建模）？有没有针对性的改进，是否有更巨大的发展空间？*

A2： 多轮任务带来的新难点和挑战在于前后轮次之间的上下文关联，用户可能因为上一个问题而在第二个问题中使用代词省略或者突然改变主题等。多轮对话中的上下文依赖就成了一个巨大挑战。模型需要理解共引用和省略，还要防止用户焦点变化时不相关的信息集成。在这方面使用 CQR(对话式问题重构) 技术，将用户该轮的问题，结合上下文进行重构，将其转化为单轮任务再结合单轮任务的优秀模型或许能有较大的改进。

Q3：评价指标方面：印象里你提到有一些 *with value* 表现好的模型在 *match* 指标中反而较差，这本身是不是一个比较大的问题？

A3：with value 注重的是 SQL 语句运行出的结果是否正确，而 match 指标是生成的 SQL 语句与标准 SQL 语句进行比较，此部分的提出可以平衡 with value 指标假阳性误判的问题。此外在论文 Dr.Spider 中使用的是 with value 作为主要评价指标，文章认为 “with value 指标评估 SQL 值的正确性，这是模型鲁棒性的重要组成部分”。我认为一些 with value 表现好的模型在 match 指标中表现差可能是因为模型本身生成 SQL 语句时不考虑 SQL 语句的风格，如 RESDSQL 模型，其在生成部分就属于偏向模板化，因此生成的 SQL 语句一般会与 gold SQL 有一定差距。从实际应用方面来说，with value 显然更受关注，至于 with value 与 match 指标差距过大，我认为其本身不是一个比较大的问题。

Q4：总体上看，预训练模型比如 *T5*，是肯定要用的基础模型，*PICARD* 用来对结果进行验证，也是必然的框架，你觉得这个框架还缺少些什么吗？这个框架本身还有可以进一步改进的地方吗？

A4：从多模态的角度出发，我认为这个框架在对数据库 schema 和自然语言问题两者之间的模态融合部分较为欠缺，可以借鉴如 CLIP、ALBEF 等双塔结构模型，为数据库 schema 和自然语言问题单独设置初步 encoder 单独编码，后续再使用一个 encoder 对单独编码后的信息进行融合，后续的解码器和 PICARD 规范不变。

References

[1] Gan Y , Chen X , Huang Q , et al. Towards Robustness of Text-to-SQL Models against Synonym Substitution:, 10.48550/arXiv.2106.01065[P]. 2021.

[2] Deng X , Awadallah A H , Meek C , et al. Structure-Grounded Pretraining for Text-to-SQL[J]. 2020.

[3] Gao C, Li B, Zhang W, et al. Towards Generalizable and Robust Text-to-SQL Parsing[J]. arXiv preprint arXiv:2210.12674, 2022.

[4] Gan Y, Chen X, Purver M. Exploring underexplored limitations of cross-domain text-to-sql generalization[J]. arXiv preprint arXiv:2109.05157, 2021.

[5] Shaw P, Chang M W, Pasupat P, et al. Compositional generalization and natural language variation: Can a semantic parsing approach handle both?[J]. arXiv preprint arXiv:2010.12725, 2020.

[6] Parthasarathi S H K, Zeng L, Hakkani-Tur D. Conversational text-to-SQL: An odyssey into state-of-the-art and challenges ahead[J]. arXiv preprint arXiv:2302.11054, 2023.

[7] Pi X, Wang B, Gao Y, et al. Towards robustness of text-to-SQL models against natural and realistic adversarial table perturbation[J]. arXiv preprint arXiv:2212.09994, 2022.

[8] Chang S, Wang J, Dong M, et al. Dr. Spider: A Diagnostic Evaluation Benchmark towards Text-to-SQL Robustness[J]. arXiv preprint arXiv:2301.08881, 2023.

[9] Li J, Hui B, Cheng R, et al. Graphix-T5: Mixing Pre-Trained Transformers with Graph-Aware Layers for Text-to-SQL Parsing[J]. arXiv preprint arXiv:2301.07507, 2023.

[10] Li H, Zhang J, Li C, et al. Decoupling the Skeleton Parsing and Schema Linking for Text-to-SQL[J]. arXiv preprint arXiv:2302.05965, 2023.

[11] Fu Y, Ou W, Yu Z, et al. MIGA: A Unified Multi-task Generation Framework for Conversational Text-to-SQL[J]. arXiv preprint arXiv:2212.09278, 2022.

[12] Xiao D, Chai L, Zhang Q W, et al. CQR-SQL: Conversational Question Reformulation Enhanced Context-Dependent Text-to-SQL Parsers[J]. arXiv preprint arXiv:2205.07686, 2022.

[13] Li J, Selvaraju R, Gotmare A, et al. Align before fuse: Vision and language representation learning with momentum distillation[J]. Advances in neural information processing systems, 2021, 34: 9694-9705.

[14] Devlin J, Chang M W, Lee K, et al. Bert: Pre-training of deep bidirectional transformers for language understanding[J]. arXiv preprint arXiv:1810.04805, 2018.

[15] Liu Y, Ott M, Goyal N, et al. Roberta: A robustly optimized bert pretraining approach[J]. arXiv preprint arXiv:1907.11692, 2019.

[16] Yin P, Neubig G, Yih W, et al. TaBERT: Pretraining for joint understanding of textual and tabular data[J]. arXiv preprint arXiv:2005.08314, 2020.

[17] Roberts A, Raffel C, Lee K, et al. Exploring the limits of transfer learning with a unified text-to-text transformer[J]. 2019.

[18] Radford A, Narasimhan K, Salimans T, et al. Improving language understanding by generative pre-training[J]. 2018.

[19] Scholak T, Schucher N, Bahdanau D. PICARD: Parsing incrementally for constrained auto-regressive decoding from language models[J]. arXiv preprint arXiv:2109.05093, 2021.

[20] 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[J]. arXiv preprint arXiv:1809.08887, 2018.

[21] Yu T, Zhang R, Yasunaga M, et al. Sparc: Cross-domain semantic parsing in context[J]. arXiv preprint arXiv:1906.02285, 2019.