Text-to-SQL 泛化性总结与多模态领域调研

日期：2023 年 4 月 6 日

摘　要

本文首先阐述我对 Text-to-SQL 领域泛化性的总结性理解，并后续在 Spider-dk 数据集中与对比验证，同时在过程中发现 Spider-dk 数据集可能存在的不严谨之处。下一部分总结了调研的图文多模态领域[1]中可以在 Text-to-SQL 领域参考的点，文章分别从模型架构和训练方式两方面阐述。最后在文末总结两部分的内容。

关键词: 泛化性，图文多模态领域

1　关于泛化性的总结

泛化性指模型经过训练后，应用到新数据并做出准确预测的能力。泛化的负面的就是过拟合现象，在训练样本上效果良好，但在新鲜样本上表现极差，这种就是泛化性级差的表现。在 Text-to-SQL 领域，泛化性想解决：**训练出的模型通过在特定数据库或者训练集上，以“取巧”的方式生成正确的 SQL 语句，而不是真正理解了问题之后的回答的问题，并辨别出此类过于乐观的模型**。为了解决这个问题，一是要防止模型“取巧”，二是要让模型对人类语言有深刻的理解[2]，三是要模型对数据库 schema 也有深刻的理解。

1.1　Spider-dk 数据结构

Spider-dk 数据集中有两个 json 文件：table.json 和 Spider.json。

table.json 中包含多个数据库 Schema 的信息数据。对于一个数据库，其中有‘column_names’、‘column_names_original’、‘column_types’等信息，如图 1 所示。其中‘db_id’为该数据库的名字“new_orchestra”。在‘column_names’与‘column_names_original’中，前者为后者中每个列对应的描述性列名，后者为数据库中列原本的名称，从多模态的角度出发，前者多在文本端作为输入，后者多在数据库 Schema 端作为输入。此外，一个数据库中含有多个表，这些表在‘table_names_original’中被列出，其与‘table_names’的关系同理。在‘column_names’中数据为所有表的列名，列名与所属表在‘table_names’中的索引组成一个列的信息。代表主键和外键的‘primary_keys’和‘foreign_keys’中的元素则为‘column_names’中对应的索引。‘column_types’用于表示列中变量的属性，包含分别表示字符串类、数字类和布尔类的‘text’、‘number’和‘others’，顺序对应‘column_names’的列顺序。

Spider-DK.json 中包含一条条训练样本，分别有‘type’、‘db_id’、‘query’和‘question’四对数据，分别指带嵌入的领域知识类型、数据库 id、正确的 SQL 语句以及 NL 查询问题。

[1]VQA 是图文多模态领域一个下游任务，通常少有单独研究 VQA 任务的文章，本次调研着重于多模态的模态对齐部分

[2]这一点可以通过在模型训练阶段给模型大量人类语言数据即可，当前的大模型如 T5，GPT3 等都已经对人类语言有足够深刻的理解

```
{
    "column_names": [ ···
    ],
    "column_names_original": [ ···
    ],
    "column_types": [ ···
    ],
    "db_id": "new_orchestra",
    "foreign_keys": [ ···
    ],
    "primary_keys": [ ···
    ],
    "table_names": [ ···
    ],
    "table_names_original": [ ···
    ]
},
```


图 1: table.json 中数据库存储形式

1.2 Spider-dk 中的五种领域知识

在 Spider-dk 数据集中，作者定义了五种领域知识 T1-T5：

T1：需要模型理解用户通过省略表达的问题。如例子“**NL**：Find the name of the ....; **SQL**: select firstname , lastname fron ....”这里就要求模型理解问题中‘name’指代‘first name’和‘last name’的含义，而不是直接在数据库 schema 中匹配‘name’这列数据。

T2：需要模型对用户问题有一定推断能力。如例子“**NL**: ...order of their date of birth from old to young; **SQL**: ...order by data_of_birth asc”这里就要求模型能做到从 from old to young 到 data_of_birth 数据递增的推理，也就是理解了生日和年龄之间的联系。

T3：需要模型能辨别同义替换问题。如例子“**NL**: ...id for French singers; **SQL**: ...from singer where country = 'France' ”即理解 French 和 France 之间的含义关系 (理解 French singers 的国籍——France)。

T4：需要模型能区分一列数据是否为布尔值类型或仅有几种值。如例子“**NL**: How many students got accepted...; **SQL**: ...where decision='yes' ”即需要模型除了知道 accepted 与 decision 之间的含义关系外，还需要知道 decision 列数据为二元变量，且取值为 yes 或者 no。

T5：需要模型能应对一些对于模型来说容易出现冲突的问题。如例子“**NL**: ...with max speed higher than 1000; **SQL**: ..where max_speed > 1000”，文中提到对于很多模型容易理解为生成 max(max_speed) 的语句，即使这点暂时没有较好的原因解释。

在 Spider-dk 数据集中，除了上述 5 类还有一些未加入任何领域知识修改的数据，以及极少的同时包含 T2T3 或者 T3T4 的数据。

总的来说，Spider-dk 数据集也是针对避免模型“取巧”，要求模型具备自然语言的相关知识，且能理解数据库中各列数据的含义设计的。对于自然语言的深刻理解在于 NL 问题中的同义替换或者简单文字推理等方面；对于数据库中数据的理解体现在需要理解不同表之间的关系以实现多表查询，以及理解列数据性质以合理取值或者设置判断语句等。

1.3 对 T4 类型问题的思考

T4 中提到模型辨别数据库 Schema 中列数据是否为 N 元变量，以及能够生成正确的值的能力。在 Spider-dk 数据集中，对于数据库 Schema 的列仅使用 text、number、others 区分类别。2.2 节 T4 中提到的例子中，decision 为 text 类别，而实际其为 N 元变量其中一个取值为'yes'。在面对一个新的数据库，如

果也遇到了此类 N 元变量，在生成 SQL 语句时会因为不知晓其中的取值，尤其是 NL 中使用同义词替换该值而生成错误的语句。关于这点数据集中有部分列名 (column_names_original) 如‘abandoned_yn’在描述性列名 (column_names) 中为‘abandoned yes or no’等能使模型学习到该列变量的取值，但仍然有不少如'genre is' 这类的与‘decision’一样未在描述性列名上说明 N 元变量取值的列，这会导致错误评估模型能力，进而使该数据集对于模型泛化性能力的判断偏低。在本质上，这属于是 Spider-dk 数据集的不严谨之处。

2 VQA 中可以参考的点

在调研的前沿多模态模型中，对于模态对齐方面，模型结构绝大多数使用基于注意力机制的多头交叉注意力机制，部分模型则在此基础上有部分改进。针对对齐部分的训练则有对比学习和掩码训练等方法。

2.1 模型架构方面

交叉注意后加上 Gate&Add 模块：文章 [7] 发表于 2020 年的 ICLR，使用视觉表示来增强机器翻译效果。在文本和图片两个模态融合中在多头交叉注意力机制上改进，加上 Gate&Add 模块，令原本在多头交叉注意力机制中输入为 query 模态的 H^L 以及被查询模态中 K_M, V_M，输出为:

$$\hat{H} = ATT_M(H^L, K_M, V_M)$$

文章认为图像信息在翻译预测过程中可能起到辅助作用，因此计算 $\lambda \in [0,1]$ 衡量图片对翻译的贡献度。添加了 W_λ，U_λ 两个可训练参数，计算方式为:

$$\lambda = sigmoid(W_\lambda \hat{H} + U_\lambda H^L)$$

$$\mathbf{H} = H^L + \lambda \hat{H}$$

实验结果方面，文章仅在机器翻译领域与不添加图像信息的 Transformer、RNN 等进行比较，对 text-to-sql 领域不一定有明确效果。文章 [2] 于 2022 年在此基础上改进，但在模态对齐方面未作修改。

交叉注意前依次使用 CNN，CBAM[3]提取: 文章 [4] 于 2023 年 3 月发表，提出了一种新的多模态分析模型。为了减少干扰信息并增强网络对模式之间相关信息的关注，CNN 和 CBAM 注意机制在文本特征和图像特征拼接后被添加，以提高特征表示能力。实验也证明模型性能与先进的模型相当。在 Text-to-SQL 领域，数据库 Schema 中存在较多干扰信息，NL 问题中存在较少甚至没有干扰信息，该改进方法可能对 Text-to-SQL 领域的模态融合有一定参考价值。

2.2 训练方式方面

从近期一些文章 [4–7] 来看，在模态融合方面，当前主流的有对比学习和掩码训练[4]两种方式训练模型。

对于掩码训练：在许多图文多模态数据集中，文本数据和图像数据内容上，或文本是对图像的描述，或文本只描述图片的一部分细节，或图文只在深层含义上相通 (如一首诗句与对应的意境图)，而对于 Text-to-SQL 领域，待融合的两个模态为 NL 问题与数据库 Schema，两模态之间并没有如图像文本之间的依赖关系，两者相对独立，因此我认为掩码训练并不适合 Text-to-SQL 领域模态的融合。

[3]CBAM（Convolutional Block Attention Module）是一种注意力机制，它可以有效地捕捉空间和通道维度之间的相关性，学习不同粒度的特征，并在不增加太多计算成本的情况下提高模型的性能。

[4]最初由 BERT[1] 模型提出，后续在多模态领域延伸出了屏蔽图像的一个 region 块，并通过其他模态，如文本来预测该 region 块的内容的方式

对于对比学习：NL 问题中包含数据库 Schema 的部分信息，对于例子 “**NL**: List all singer names in concerts after or in year 2014; **SQL**: SELECT T2.name FROM singer_in_concert AS T1 JOIN singer AS T2 ON T1.singer_id = T2.singer_id JOIN concert AS T3 ON T1.concert_id = T3.concert_id WHERE T3.year >= 2014; **db_id**: new_concert_singer” 可以看出,NL 问题中‘singer’，‘name’，‘concert’以及与时间相关的‘year’在数据库中与‘singer_id’列，‘year’列以及‘singer_in_concert’表，‘singer’表等存在语义联系。因此我认为对比学习对 Text-to-SQL 领域模态融合有一定参考价值。另外考虑到当前语言类型大模型发展较为迅速，对比学习中更多的可能是数据库的 Schema 结构化语言更多地向文本语言靠拢，即训练中采取类似 CLIP[3] 中冷冻文本模态部分的参数，加强视觉模态想文本模态靠拢的方式。

3 总结

泛化性想解决的问题：训练出的模型通过在特定数据库或者训练集上，以“取巧”的方式生成正确的 SQL 语句，而不是真正理解了问题之后的回答的问题，进而过滤掉准确率较高的“作弊”模型。

图文多模态领域可以借鉴的点：可以在模态对齐的方面，使用多头交叉注意力机制，且可以在该机制上添加 Gate&Add 或者使用 CNN、CBAM 等方法改进模型框架。同时，可以先使用对比学习对齐两模态的 Encoder 部分，再做后续的整体训练。

参考文献

[1] Jacob Devlin et al. *BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding*. 2019. arXiv: 1810.04805 [cs.CL].

[2] Qingkai Fang and Yang Feng. *Neural Machine Translation with Phrase-Level Universal Visual Representations*. 2022. arXiv: 2203.10299 [cs.CL].

[3] Alec Radford et al. *Learning Transferable Visual Models From Natural Language Supervision*. 2021. arXiv: 2103.00020 [cs.CV].

[4] Huiru Wang et al. *Exploring Multimodal Sentiment Analysis via CBAM Attention and Double-layer BiLSTM Architecture*. 2023. arXiv: 2303.14708 [cs.CV].

[5] Longzheng Wang et al. *Cross-modal Contrastive Learning for Multimodal Fake News Detection*. 2023. arXiv: 2302.14057 [cs.LG].

[6] Wenhui Wang et al. *Image as a Foreign Language: BEiT Pretraining for All Vision and Vision-Language Tasks*. 2022. arXiv: 2208.10442 [cs.CV].

[7] Zhuosheng Zhang et al. “Neural Machine Translation with Universal Visual Representation”. In: *International Conference on Learning Representations*. 2020. URL: https://openreview.net/forum?id=Byl8hhNYPS.