**数据集：**

**早期数据集：**ATIS（Airline Travel Information Systems，航空订票系统）、GeoQuery（地理查询系统）等都是限定于某个特定领域的数据库，仅能解决特定领域的问题，泛化性能较差。



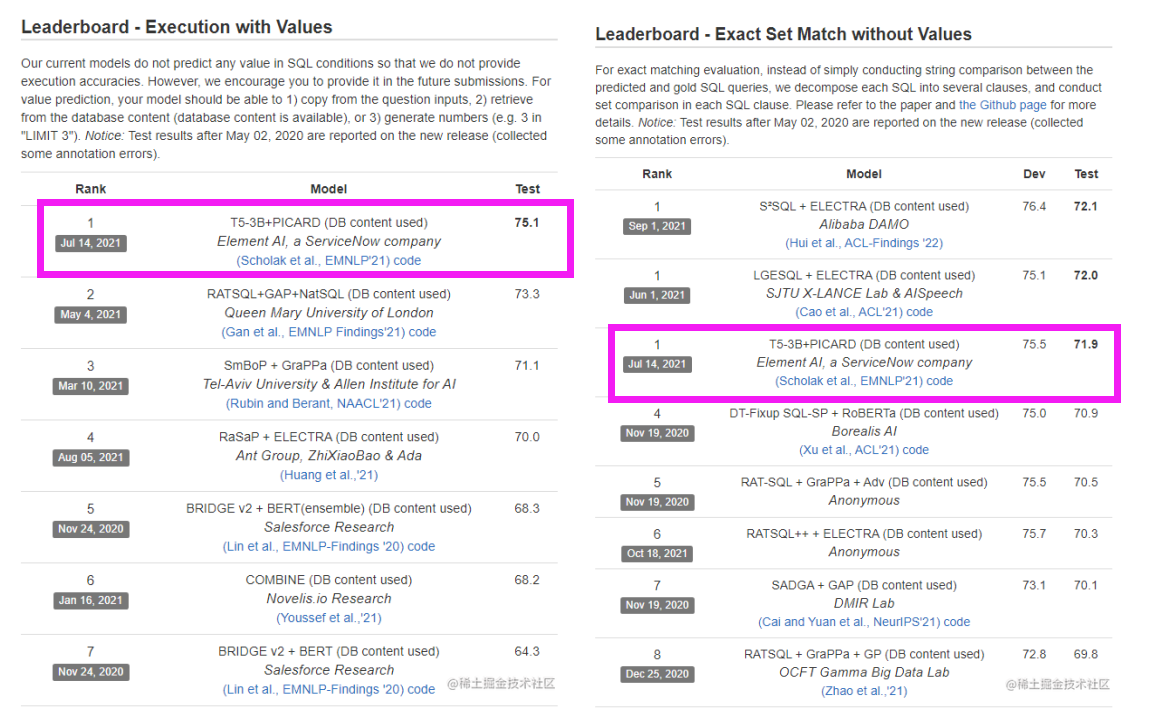
2021年ACL上还有一篇  CHASE:首个跨领域多轮Text2SQL中文数据集

**WikiSQL：**查询和表的数量很多，但是所有 SQL 查询都很简单，仅包含 SELECT 和 WHERE 从句，而且WHERE子句中的约束条件不超过3个。此外，每个数据库都只是没有外键的简单的表。在 WikiSQL 上训练的模型在无法处理复杂的 SQL语句要素（如 GROUP BY、ORDER BY 或嵌套查询）和具备多个表和外键的数据库。

**Spider：**多数据库、多表、单轮查询的Text-to-SQL数据集；包含 10181 个问题和 5693 个独特的复杂 SQL 查询、200 个具备多个表的数据库，覆盖 138 个不同领域，实际应用性很强，问题也复杂很多。相应的SQL语句中包含了JOIN、GROUP BY、HAVING、嵌套等诸多SQL高级语法内容。同时，该数据集也根据相应SQL语句的复杂程度划分了不同的难度等级。

测试集不公开，研究者需要将自己的模型在验证集上调试到最佳性能后将模型发给Spider官方，有官方在测试集上进行测试，并将结果公布在Leaderboard上以确保模型方法对比的公平性。

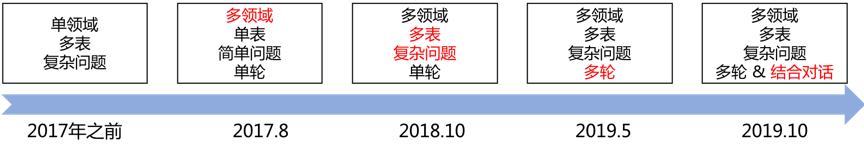
评价指标：Exact Match(EM): 模型预测的SQL语句必须与ground truth完全一样；Execution Accuracy(EX)：模型预测的语句执行后所得的结果与ground truth一样。



**一些模型：**

* **SQLNet(一个baseline)**
* **TypeSQL**

其实这两个都比较老了



**看的论文：**

对于Spider数据集：Relation-Aware Schema Encoding and Linking for Text-to-SQL Parsers->升级版本：LGESQL-Line Graph Enhanced Text-to-SQL Model with Mixed Local (最初是基于GNN的改进而来)AST生成sql，对sql理解更生动一些

对表格做预训练：TABERT: Pretraining for Joint Understanding of Textual and Tabular Data(P Yin，G Neubig et al., 2020) -> (好一点的)GRAPPA: Grammar-Augmented Pre-Training for Table Semantic Parsing->(GRAPPA升级版)SCORE 针对spider数据集人工统计的90个常用sql模板，然后多分类套出来的sql，在RoBerta上改进来的模型

上面的都是context-free类型的，下面的对话情景text2sql才是注意上下文关联的

对于多轮text2sql——IGSQL: Database Schema Interaction Graph Based Neural Model for Context-Dependent Text-to-SQL Generation 是EditSQL的改良版本，其中也使用了图形式存储schema，但是跟RAT-SQL有点不一样，个人感觉这种还行(这种还在不断改变根据NL改变graph的节点值，达到了对上下文语境的理解，这点正是本文在EditSQL上改良的重点)

而在一个临近的领域NL-PL中出现过codeBert -> GraphCodeBert的升级就是因为AST，可以见得AST的潜力。

感觉IGSQL的生成方法类似于GPT中的生成虽说不同于AST但是这种方法的潜力在chatGPT中得到了展现，即使使用AST的模型提到过“前者的生成中一旦一个token预测错误，整个sql语句的预测也就出错了”，在此也认为只要训练数据量上去了前者可以避免上述问题的。而AST本身应用于code generation效果良好，同样在SQL generation上有良好前景。

补充：对于AST：A Syntactic Neural Model for General-Purpose Code Generation

参考的一个很重要的2021年的Text-to-SQL综述：[2021你还在手写SQL吗？万字综述Text to SQL技术 - 知乎 (zhihu.com)](https://zhuanlan.zhihu.com/p/430369730)

查应用领域以及背景时的一些：[”想知道你家爱豆最近的演唱会？"让Text2SQL模型自动帮你回答！-阿里云开发者社区 (aliyun.com)](https://developer.aliyun.com/article/936285)

[语义解析 (Text-to-SQL) 技术研究及应用 下篇 - 知乎 (zhihu.com)](https://zhuanlan.zhihu.com/p/269478469)

感谢你的综述，几天时间内你整理了很多资料，做的不错！不知道是否有参考已有的综述文章？如果有的话，可能得予以引用并加以说明。这涉及学术规范问题，还是严谨为好。

我看到你在思考部分讨论了任务、架构两个方面的问题。看上去你是觉得在单轮任务中 基于图的结构表示+AST解码 会是该领域较强的架构 对吗？那目前达到75%的sota是使用这个架构吗？他们仍然存在的问题是什么呢？

另外，问个更基本一些的问题，帮助你思考：你觉得text2sql本质上难在哪里？有哪些部分我们已经可以解决，有哪些部分是难题，可能暂时还没法解决的？

单轮任务中sota模型是Graphix-T5结合PICARD，从原本decoder输出进行生成SQL。模型整体使用的Seq2Seq架构。编码部分在GNN-T5模型上改进，每一个encode layer由T5模型encode layer和图注意层(graph-aware layer)层共同编码并加和组成。解码器部分在原本的T5模型decoder基础上后缀加入了PICARD(其中使用了AST解码，但与其他AST解码的模型不同)。

该模型仍然存在的问题(个人观点): 1)编码层graph-aware layer和T5-encoder的结合不紧密，是类似concat的拼接，融合并不完善，但也取得了较好的成果，完全融合可能向GNN-T5一样出现灾难性遗忘的问题。2)根据Dr.Spider论文提出的观点，Graphix-T5模型在应对与生活应用更贴切的有扰动的数据集中可能存在进步的空间，这是目前所有模型都需要解决的问题，即模型的鲁棒性。

text2sql问题本质难在哪里：

单轮：

多轮：考虑上下文的依赖性和数据库模式结构的情况下如何巧妙地代表多轮问题和数据模式。存在上下文语义代指(eg:them),英语语法中缩写等——from CQR-SQL

decode部分：

* 大型预训练模型中decoder结构生成(T5)外加PICARD规范(其中有使用AST加以规范的步骤、这个方法单轮多轮sota都在用)——SOTA
* 根据解码器生成隐藏层生成AST语法树，并根据语法树解码生成SQL，该方法分为自顶向下(GRAPPA)与自底向下(SMBOP)，当然这种方式与PLM并不兼容。

当前需要最大限度地发挥预训练T5模型对目标任务的潜力。

可以改进的方面：

1. 多轮任务MIGA中单纯用的T5的模型加PICARD，可以在其encoder上学习Graphix-T5，对于T5-encoder输入部分是字符串直接拼接的，感觉可以改进(CQR)，原本的方法很快就会达到512个token，导致出现截断。
2. MIGA中对于预训练模型的训练任务看看可不可以优化
3. 单轮/多轮任务的评价指标可以优化
4. 对数据集进行分类，从不同能力方面训练模型(具体需要训练哪些能力得分问题)
5. CQR那个数据集一直没有公开，没法用。
6. 两个领域都适用的方法：使用优质合成(加噪音)数据再次训练

尊敬的黄教授，您好

感谢您这段时间的指导，这段时间我接触了text2sql这个有趣的领域，也练习了一些文献调研，综述写作的能力，同时我也认识到自己科研能力、科研素养方面仍有较大欠缺。未来我会继续提升自己的科研水平，希望能够保研去到贵校。

再次感谢，祝您身体健康，生活愉快！

周呈星

神经网络相关：

pytorch简单看一下，会用，知道矩阵张量这些，看一下使用GPU加速。

神经网络找B站看一个CNN(卷积神经网络的代码教学，虽然用不上CNN，但是可以学一下深度学习基本代码)

NLP相关：

transformer模型(可以看B站解说或者知乎推文)

Bert模型（以transformer的一部分encoder部分构成