面向官方统计的文本到 SQL基准

1st Lucas Lageweg阿姆斯特丹大学阿姆斯特丹，荷兰 l.lageweg@uva.nl

2nd Karen Goes荷兰统计局海牙，荷兰 kwm.goes@cbs.nl

3rd Tom van den Broek荷兰统计局海牙，荷兰 aj.vandenbroek@cbs.nl

4th Frank Pijpers荷兰统计局海牙，荷兰 f.pijpers@cbs.nl

5th Paul Groth阿姆斯特丹大学阿姆斯特丹，荷兰 p.t.groth@uva.nl

6th Jan-Christoph Kalo阿姆斯特丹大学阿姆斯特丹，荷兰 j.c.kalo@uva.nl

arXiv:2611.32646v1 [cs.AI] 24 Nov 2026

摘要—**官方统计是对政策制定者、企业等各类主体至关重要的数据集。对于官方统计数据的提供者而言，提高数据的可访问性、治理水平和披露程度至关重要。通过提供基于文本到 SQL 方法的问答系统，可以改善这一状况。然而，现有的文本到 SQL 基准测试并不符合这一应用场景。官方统计数据以开放域目录的形式发布，其中包含数千张宽表，列和取值均为代码，其含义记录在单独的元数据中，而非关系模式中。为弥补这一不足，我们推出 LOCuST，这是一个文本到 SQL 基准测试，配有来自荷兰统计局的两千多张英语和荷兰语表，以及一个语义知识图谱。该图谱通过标签、定义、层级和单位描述这些表、度量和维度代码，是系统查找正确表并将问题对应到正确代码所需的外部知识。该基准测试包含专家标注的问题，涵盖不同的查询和聚合类型，包括原始值检索、求和、比例计算和连接。除标准评估指标外，我们还提出了观测值 F1 指标，用于衡量查询与真实值的接近程度，即使两者答案不同也能进行评估。我们评估了多种最先进的大语言模型，发现它们难以编写能够正确回答问题的 SQL。**

索引词—**文本到 SQL、基准测试、官方统计、大语言模型、问答、知识图谱、表检索、多语言**

I. 引言

官方统计（如人口普查和经济数据）是政策制定者和企业依据事实进行决策的基础 [1], [2]。然而，在大量统计数据中查找特定用例的相关数据可能既困难又耗时。比如，用户需要预先了解有哪些数据、可以在哪里找到这些数据，以及数据是如何组织的。因此，生成此类数据集的机构致力于为终端用户提供便捷的访问方式 [3]。一种备受关注的解决方案是通过文本到 SQL 方法，对统计数据提供自然语言问答 [4]。这些方法将自然语言问题转换为可在底层数据库上执行的 SQL 查询。

官方统计涉及描述复杂现实世界现象 的事实，也可能包含并非自我解释的概念，例如（国际）标准行业分类（SIC）、不平等程度的统计离散度（基尼指数），或涵盖各种商品和服务的消费者价格指数（CPI）。官方统计在描述概念时也会使用领域专用行话和法律术语，例如“围产期死亡率”（死产和出生后不久死亡）、“购买力平价”（用于换算不同价格水平的货币）和“阿米巴病”（由寄生性阿米巴感染导致的死亡原因）。国家统计机构通常使用各自国家的语言提供统计数据，因此，以英语为主的基准测试更无法充分代表这类情况。此外，含义模糊的代码、属性名称和复杂标识符需要额外的元数据，才能正确回答问题。在官方统计中，这些外部知识通常会以定义的形式记录下来，或通过建模纳入元数据中。Bouattour 等人 [5] 表明，模式语义对企业级文本到SQL至关重要，因为仅凭原始模式信息，很难理解缩写名称和领域专用术语。最后，从表中查询和汇总值时，了解这些度量的计算方法和单位对于给出正确答案至关重要。

遗憾的是，现有的文本到 SQL 基准测试无法反映统计数据的复杂性。例如，Spider [6] 和 BIRD [8]基于通常可获得的网络抓取数据、广泛存在且长期使用的数据集，和/或仅涵盖少量领域的数据，或涵盖互联网上最常见的领域（如体育）的数据，
电影和书籍以及零售数据库。其他较新的文本到 SQL 基准测试（如 Spider2.0 [7]和 BEAVER [10]）改进了这些局限，但同样未针对目录级检索进行基准测试，并将范围限制在单数据库封闭领域生成。TACO [12] 是一个例外，因为它测试了目录级检索。现有的大多数文本到 SQL 基准测试都依赖模型解读数据库模式并从中推导知识，以回答问题。这些基准测试均不包含由数据发布者自行提供的原生语义元数据，也不会验证输出的语义有效性。一些基准测试（如 BIRD）要求使用“外部知识”来回答问题——

表 I：热门文本到 SQL 数据集的统计信息，* 展示查询数量、表数量、每个表的平均列数，以及各数据集对使用外部知识这一要求的覆盖情况。

数据集	查询数	表数**	平均值 每表列数	中位数 # 列/表	语言	外部知识
Spider [6]	10,181	995	5.10	4	en	✗
Spider2.0-snow [7]	547	5,137	28.11	10	en	外部文档
BIRD [8]	10,962	597	7.26	5	en	证据句
BULL [9]	4,966	78	14.96	13	英语、中文	模式描述
BEAVER [10]	7,978	381	9.50	7	en	领域知识标注
StatBot.Swiss [11]	456	36	6.94	5	英语、德语	模式描述
TACO-Beijing [12]	6,354	2,211	15.83	6	zh	✗
TACO-US [12]	5,074	9,872	19.40	11	en	✗
LOCuST-en	1,299	1,030	32.40	12	en	知识图谱
LOCuST-nl	1,260	1,211	34.55	10	nl	知识图谱

* 所有统计数据均基于各数据集的公开资源。数据集统计分析截至2026年9月。部分数据集自首次发布以来可能已更新，且确实有过更新。**按数据库统计表的数量，包括按时间段拆分的表（例如每年一张表），以及出现在多个数据库中的表。

然而，这些知识是非结构化的，并且针对特定问题进行了标注。所需知识也会在评估期间提供给模型。

为弥补这些不足，我们推出 LOCuST 基准测试（**大规模官方统计复杂**文本到**SQL任务**）：这是一个大规模、真实场景的文本到 SQL 数据集，包含有关复杂官方统计的英语和荷兰语问题。[1] 该数据集由英语部分和荷兰语部分组成，英语部分包含 1,030 张表上的 1,299 个问题-查询对，荷兰语部分则包含 1,211 张不同的表上的 1,260 个问题-查询对。每个部分分别选取 308 个（英语）和 299 个（荷兰语）问题作为测试集。该基准测试提供了一个具有挑战性的新环境，用于测试文本到 SQL 系统。我们的贡献如下：

- 一个全新的、由专家标注的文本到 SQL 基准测试，基于涵盖 22 个领域的真实世界官方统计数据，包含 2,241 张表和 2,559 个英语及荷兰语问题。该基准测试涵盖了从小型表（$<$ 5 列）到超大型表（$>$ 1，200 列）的各种模式，反映了统计实践中遇到的表宽度长尾分布。

- 一个分为三个阶段的任务环境，用于测试目录级检索和开放领域文本到 SQL 生成。
- 与这些表配套的一个庞大语义知识图谱，其中包含所涉及概念的标签、描述、单位及其他相关元数据，用于验证不同基准测试任务中答案生成的语义依据。

- 一种新的观测值 F1 指标，可用于获取相关洞见，并与现有指标相比评估文本到 SQL 性能。

- 实验表明，即使所提查询并不复杂，当前最先进的大语言模型在处理我们基准测试中呈现的现实世界概念时仍然表现不佳：端到端观测值 F1 值最高仅为 0.62（英语和荷兰语；GPT 6），数值 F1 值为 0.52（英语；GPT 6），在我们的基准测试中，执行准确率为 0.10（英语；GPT 6）和 0.05（荷兰语；Qwen 3.8）。

[1]所有资源均可在 https://github.com/lagewel001/LOCuST 获取

II. 相关工作

文本到 SQL 数十年来一直是备受关注的研究课题 [4], [13]。大语言模型的引入显著提升了系统性能，也进一步吸引了人们对该任务的关注。由此产生了各种基准数据集，我们将在此进行综述。表 I 展示了常见文本到 SQL 数据集的比较。对于每个数据集，我们统计了所有公开发布的问题-SQL 对，以及每个数据库中的所有表，但不包括未公开的测试集。

Spider [6] 和 BIRD [8] 是大型数据集，包含众包、网络抓取和合成数据及其标注。虽然它们为测试 SQL 查询生成提供了有力的基准，但其中包含的表较为狭窄（列数中位数为 4–5），且数据和领域本身更加通用，涵盖面向大众的一般性公共领域。这无法反映统计学或企业场景中数据的复杂性，因此也无法对这类复杂性进行基准测试。

近期发布的新基准测试着重考察企业数据和工作流程，结果表明，文本到 SQL 模型要应用于这些真实场景，仍需进一步改进。BULL [9],BEAVER [10]和 Spider2.0 [7] 是这类企业级文本到 SQL 数据集的例子。与前面提到的众包替代数据集相比，这些数据集包含大语言模型较不熟悉领域中的企业数据（已匿名化）。然而，BULL 和 BEAVER 涵盖的领域范围有限：BULL 仅涉及金融数据，BEAVER 涉及 19 个领域，但这些领域仅来自三家组织的业务数据库。此外，与 Spider 和 BIRD 一样，这些数据集中的每道基准测试题都带有数据库标识符，将每道题限定在一个数据库中。尽管在技术上涵盖多个领域，这些数据集仍是在封闭域设置下进行基准测试，每道评估题只提供一个

数据库。LOCuST 涵盖某一出版商的 22 个领域，并以目录规模的开放域检索为基准测试对象，使文本到 SQL 不再是仅生成 SQL 查询的孤立任务，而是对系统完整端到端检索能力的考验。

近期新增的开放域文本到 SQL 基准测试还包括 TACO [12]，，该基准测试基于来自北京和美国政府门户网站的 46 个开放数据数据库中的 12,000 个表，提供了大量合成查询。TACO 将在其完整目录中寻找正确的表作为任务的一部分。然而，TACO 并未涵盖供模型在生成 SQL 时使用的外部知识，也未涵盖其跨数据库开放域设置基准测试时的语义有效性和连贯性。跨数据库问题无法确保连接的数据在语义上相关。在 TACO-北京子集中，756 个跨数据库连接条件中有 586 个将两个不相关表的行号列进行等同（例如，SequenceNumber = SequenceNumber），因此其结果在任何语义层面都没有意义。相比之下，LOCuST 将目录级检索与针对宽型、非规范化统计表的专家标注问题相结合。

BIRD、Spider2.0、BEAVER 和 BULL 都以某种形式提供了外部知识和模式语义（见表 I）。Bouattour 等人 [5] 表明，提供良好的模式语义对于生成有意义且正确的 SQL 非常重要。对于以某种形式提供这种外部知识的数据集而言，这些知识是非结构化的，通常由人工按问题或按数据库进行标注，然后提供给模型。LOCuST 为整个目录增添了一个结构化知识图谱，其中包含标签、定义和单位。它既用于建立问题的语义依据，也用于验证答案，并且在端到端文本到 SQL任务中必须被检索。

StatBot.Swiss [11] 是一个官方统计文本到 SQL 数据集，与此处介绍的使用场景最为接近。它包含来自瑞士开放政府数据门户、由专家整理的英语和德语问题。不过，该数据集的规模和模式复杂度有限：它包含针对预先规范化的窄型事实表提出的 456 个问题（中位数为 5），而此处使用的是宽得多的非规范化 OData 表（中位数为 10–12）。与 StatBot.Swiss 中每个数据集仅以一种语言发布不同，荷兰统计局目录中相当一部分内容同时提供荷兰语和英语版本，且使用相同的表标识符，因此可以在对齐的子集上进行受控的跨语言比较。此外，该数据集不支持端到端生成，即在查询生成流程之前加入表检索任务。缺少目录级检索限制了该数据集在真实世界使用场景中的适用性。

与现有数据集的差异：尽管上述数据集对于为文本到 SQL 系统的进一步改进奠定基础至关重要，但仍需要一个新的基准测试来应对统计数据的关键挑战。与先前的文本到 SQL 基准测试相比，LOCuST 的不同之处在于：(1) 模式规模：LOCuST 每种语言包含一千多张表，对于每个问题-SQL 对，这些表均被视为范围内，并且包含大量宽表，从而增加了列选择和表选择中的歧义；(2)多领域统计数据：这些表涵盖22个领域，并经常使用含义并不自明的标识符和领域专用术语，仅凭模式难以解析；(3)多语言性：该基准测试同时涵盖英语和荷兰语，并反映官方统计报告中使用的语言；(4)结构化外部知识：LOCuST 并非主要依赖非结构化文档，而是提供一个语义元数据知识图谱，其中包含度量与维度的标签、定义、描述和单位；(5)数据驱动的难度：回答 LOCuST 问题需要稳健的依据与消歧（例如解析代码、定义和单位），即使所需的 SQL模式相对标准；以及 (6) 目录级检索：LOCuST 还在大规模上引出了一个困难的表检索问题。因此，该基准测试包含检索基线，并在提供相关表时额外评估查询生成，以分离语义落地和 SQL生成。所讨论的现有基准测试（TACO 除外）为每个问题包含一个数据库标识符，从而在针对某个问题进行 SQL生成时，将推理范围限制为单个数据库。我们将整个目录视为范围内，并将其作为信息检索任务的一部分。

III. 基准测试

荷兰统计局[2] 是荷兰政府的国家统计机构，提供关于荷兰和加勒比荷兰（博奈尔、圣尤斯歇斯、萨巴）的各类统计数据。大多数统计数据以荷兰语和英语提供，并依据《荷兰统计局法》（《荷兰中央统计局法》） [14]向公众开放。荷兰统计局提供的表以遵循非规范化列式结构的开放数据第3版规范[3], 存储。数据集中的表包含统计观测值，这些观测值通过度量 （即被测量并产生观测值的概念）和一个或多个维度 （即变量）进行索引。表 IIa 展示了一个示例，说明不同学校类型的在籍学生人数。每个（绿色）列都包含与相应维度相关的代码，而每个度量列则包含数值观测值（蓝色），表示给定行中维度代码对应的度量值。这种结构在概念上类似于数据仓储中的星型模式，其中维度表通过外键关系连接事实表（在此案例中相当于观测值）。在我们的案例中，“事实”周围的维度和元数据由语义知识图谱提供。

为了展示端到端的示例，我们可以考虑自然语言问题 *‘2003*年和*2004*年期间，罗马天主教小型小学的平均学生人数是多少？’。用于生成 SQL 的函数或模型将会是

[2]https://www.cbs.nl/英语-gb [3] https://www.odata.org/documentation/odata-version-3-0/odata-version-3-0-core-protocol/

表 II：OData3 表和查询结果示例。

教育类型	学校规模	时期	在校学生人数
A025301	A041907	2022SJ00	98213
A025301	A041907	2023SJ00	98213
A025301	A041907	2024SJ00	99221
A025301	A041909	2022SJ00	61198
A025301	A041909	2023SJ00	54136

(a) 示例表格，其中维度组以深绿色显示，维度代码以浅绿色显示，度量以橙色显示，观测值以蓝色显示。所有标识符都通过知识图谱中的标签和定义进行描述。例如，学校规模 → *A041907* 的定义是拥有*101*至*201*名学生的机构。

教育类型	学校规模	意识形态基础	AVG
小学	少于 101 学生	罗马天主教	13755.0

（b）示例 1 的查询结果，其中对各时期的度量值（橙色）取了平均值。颜色表示维度代码（浅绿色）和观测值（蓝色）。标签取自语义图。

给定这个自然语言问题和语义元数据知识图谱。图1展示了与示例问题相关的知识图谱示意图。该函数可以访问包含所有表的完整知识图谱；但是，为了生成正确的 SQL 查询，它必须先检索相关部分。在我们的函数从知识图谱中选出用于回答问题的相关表后，必须利用这些信息生成有效的 SQL 查询。我们函数的预期结果是示例 1 所示的 SQL 查询。SQL 查询通过基准测试执行并验证后，最终查询结果（包括转换为可读标签的相关代码）如表 IIb 所示。[4]

```
选择 教育类型、意识形态基础、学校规模，平均值(在校学生人数_2)
```

示例 1：问题的预期 SQL 查询：‘*2003*年和*2004*年，小型罗马天主教小学的平均在校生人数是多少？’

所有基准测试数据均以 Parquet 文件形式提供，分为英语和荷兰语两个独立数据集。我们使用 DuckDB 作为离线访问和查询数据集的基础工具。

为测试领域迁移并避免训练集与评估集之间发生领域泄漏，我们通过将整个领域留出，构建文本到 SQL 划分。我们将所有答案需要用到教育、能源、国际贸易、制造业、自然与环境以及安全与司法领域中表的问题划入测试集，并使用其余领域的问题进行训练。重要的是，在两个划分中，模型都可以访问完整的 1,000+ 个表的集合。划分仅改变问题所属的领域。

[4]答案中代码的实际标注仅用于分析时提高可读性，不会影响基准测试。

这些问题涵盖 8 种不同的查询类型，我们按照难度顺序列出如下：

- 值，要求从单个表中获取一个简单的值；
- 求和与平均值，要求对所有查询的度量或某个维度进行求和或计算平均值；
- 最小值与最大值，要求在所有查询的度量或某个维度中找出整体最小值或最大值；
- 连接，要求从多个表中获取多个值，这些表通过某个度量或维度连接；
- 属性，要求获取给定度量或维度相对于其总值的比例；
- 聚合连接，要求对多个连接的表应用某个聚合函数。

A. 语义元数据图

该数据集附带一个内容丰富的元数据知识图谱，其中包含描述表、度量和维度的信息。对于数据集中的所有概念，图谱会在有相关信息时提供标签、定义、描述和单位。模型可以利用这些信息查找正确的概念，从而准确回答问题。

数据库模式可能包含含义不明确的列名或无法用单个术语描述的复杂概念。此外，仅凭数据库模式无法表达计算方法、不同的定义以及概念之间的关系。例如，“绿色能源”这一维度在一个表中可能将生物质燃料作为一种能源纳入，但在另一个表中却将其排除；某个市镇在某一年可能作为独立实体存在，但在之后的一年可能与更大的市镇合并；或者某个度量的计算方法可能随时间发生变化，荷兰消费者价格指数中的电价计算就是如此，该计算方法在2023年发生了变化。包含这些信息的知识图谱为模型检索和使用外部知识提供了一种结构化方式。

该知识图谱结构由人工根据荷兰统计局通过所提供的 Open Data 3 API 提供的数据构建而成。其核心是使用 Data Cube 词汇表 [15]（不包括观测值），将数据构建为包含表、度量和维度节点的数据立方体。度量和维度之间的层级关系，以及概念标签和描述，均使用 SKOS 词汇表 [16]进行建模。在描述表元数据时，使用了 Data Catalog 词汇表 [17] 和 Dublin Core 本体 [18]。对于度量单位的建模，在可能的情况下，将原始度量中的单位映射到 QUDT 本体 [19]。

图1展示了一个表子图的示意示例，其中包含与其相连的度量和维度及其层级关系。颜色对应于表 IIb 中的查询答案示例。需要注意的是，该知识图谱仅包含表所涵盖概念的元数据，而实际观测值只能在数据表本身中找到。

该图用于在使用框架执行查询时，将表中的代码转换为相应的标签。

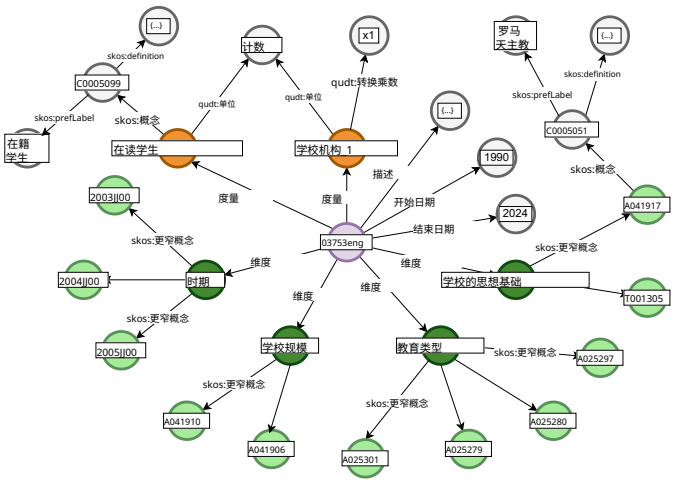

图1：表03753eng（按教育类型和意识形态基础划分的学校规模）语义图的示意概览。为简洁起见，此处仅展示了部分图对象和谓词。所有度量和维度均具有可提供给模型的带标签定义。度量以橙色突出显示，维度组以深绿色突出显示，相关联的维度代码以浅绿色突出显示。

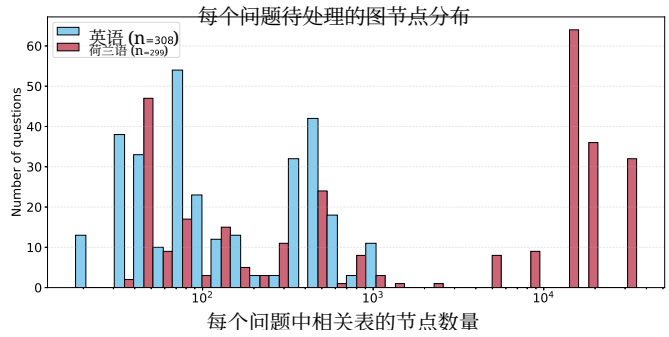

图2：每个问答对的相关表中需要搜索的度量 +维度数量分布。

此外，基准测试框架还使用该图来评估查询中聚合的有效性。对于单位不同或没有单位的度量（例如百分比、计数、比率、人均国内生产总值等），其查询中的聚合操作不被允许，并视为错误。例如，模型不能对两个不同度量的百分比求和，也不能将权重与货币观测值相加。第四节将进一步介绍元数据图如何序列化，以用于我们的文本到SQL 任务。

表III展示了元数据图的规模和连接数量的一些统计信息。利用该图，我们可以分析与每个表相连接的度量和维度数量。图2展示了英语和荷兰语测试集中每个问题—答案对所涉及的相关表中度量和维度的分布情况。可以看出，对于荷兰语测试集，其问题的相关表包含

表III：两种语言的元数据图统计

	英语	荷兰语
三元组总数	1,701,066	6,531,863
表	1,030	1,211
包含以下内容的表 *geo dim.*	139	362
包含时间维度的表	1,012	1,068
唯一 度量	21,157	15,400
度量-表链接	29,886	36,816
带有 QUDT 单位的度量	80.6%	93.5%
出现在多个表中的度量	21%	40%
唯一维度	435	604
表-维度连接	2,468	3,726
维度代码	32,737	88,169
独特概念	52,380	166,229

显著更多的维度和/或度量，而英语测试集则相对较少。这主要是因为荷兰语数据集包含了更多关于邻域层级的统计信息，而英语数据集并非如此。除了作为一种低资源语言，其文本到SQL任务更具挑战性之外，需要处理的维度数量更多，也使得荷兰语数据集相较于英语数据集更具挑战性。在比较两种语言的结果时，应将这一点纳入考量。

在以下各节中，我们将介绍基准任务，并深入说明数据集中的表和查询。

B. 任务表述

文本到 SQL 的目标是将自然语言问题 N 转换为 SQL 查询 Q。在数据库 $S = \langle \mathcal{T},\ C \rangle$上执行时，其中 $\mathcal{T}$ 和 C 分别表示可用表集合及其对应的列，查询应得出问题的正确答案。此外，还提供了语义图 $G = \langle \mathcal{T},\ \mathcal{M},\ \mathcal{D} \rangle$，其中 $\mathcal{T}$表示可用表的集合，而 $\mathcal{M}$ 和 $\mathcal{D}$ 是 C 的不相交子集，分别表示度量和维度。我们将 LOCuST 分为三项任务，每项任务均由一个将特定输入映射为所需输出的函数定义（另见图 3）：

- 表检索：针对给定问题检索正确且相关的表，以便用于回答该问题。此任务可由一个函数完成

$$R = f_r(N, S, G)$$

  其中，输出 $R \subseteq \mathcal{T}$表示与问题 N相关的（按排名排序的）表列表。
- 查询生成：给定与答案相关的正确表，生成 SQL 查询；此任务可形式化为函数

$$\hat{Y} = f_q(N, \mathcal{T}', G)$$

  其中，输入 $\mathcal{T}' \subseteq \mathcal{T}$是回答问题 N所需的相关表子集，输出是 SQL 查询 Q，它会返回问题的正确答案，并在查询输出中包含用于回答该问题的所有度量和维度。请注意，只有

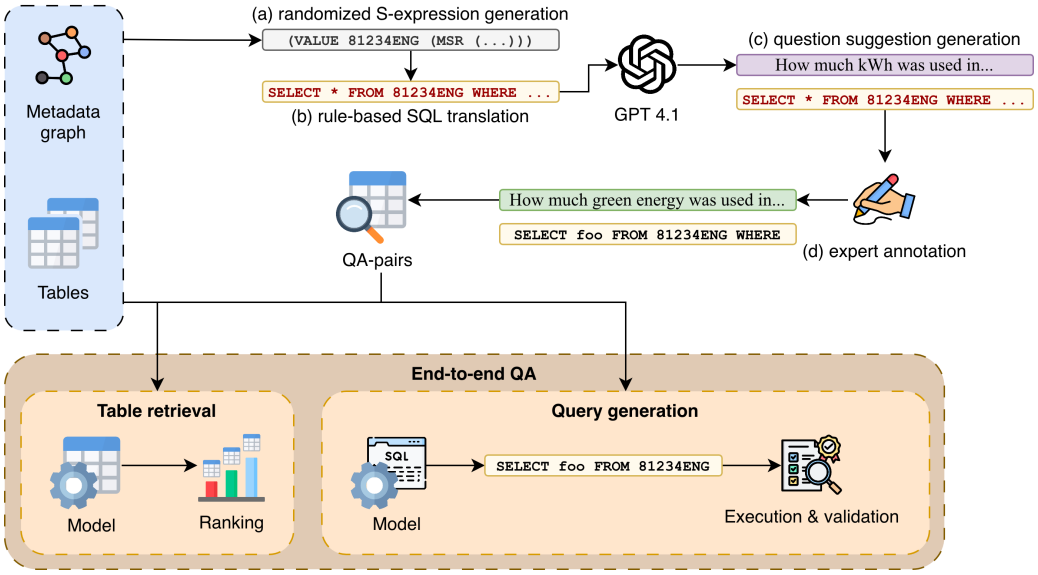

图 3：基准测试中标注工作流和所涉及任务的概览。

给出了相关表，因此在生成 SQL 时仍需执行模式链接。

- 端到端问答：结合表检索与查询生成，即针对给定问题生成查询，而不预先提供正确的表。该过程由一个函数完成

$$\hat{Y} = f_e(N, S, G)$$

该任务可以由单个单体模型解决，也可以由不同模型分别完成表检索和查询生成，使得 $f_e = f_q(N,\ f_r(N,\ S,\ G)\ G)$。

C. 标注

除 SQL 外，该数据集还包含第二种查询类型：定制的 S 表达式，灵感来自 Gu 等人的研究 [20]；这些表达式可以更容易地与原生开放数据查询相互转换，这对于在线环境中的数据查询很重要，同时也用于标注工作流。在这里的基准测试中，我们只关注 SQL 查询；但对于感兴趣的读者，数据集提供了 S 表达式及其规范。

为创建问答对（QA-pairs），系统根据查询类型、表及其对应的维度和度量的基于规则的随机选择，自动生成 S 表达式（图 3a ）。随后，系统使用模板和基于规则的系统，将这些 S 表达式转换为对应的 SQL（图 3b）。对于每个生成的查询，系统向 OpenAI GPT 4.1 提供 S 表达式、图中的相应元数据以及执行查询的结果，以生成问题建议（图 3c）。随后，荷兰统计局的四位专家人工检查每个生成的问题建议及答案对，并在必要时予以接受、拒绝、调整或重写。根据具体情况，可以修改问题或查询（图 3d）。

在标注过程中，研究人员注意到，与荷兰语相比，GPT 4.1 为英语查询提供的问题建议可靠得多。标注人员必须确保问题中提及的所有概念都由查询体现，反之亦然。对于荷兰语，这一点往往无法满足；或者，系统会生成询问所提供答案中未给出的关系的问题。这使得与英语数据集相比，荷兰语问答对需要更多人工调整或重写。

D. 指标

本节定义用于评估模型性能的指标。我们既纳入了文本到 SQL 基准测试的经典指标，也定义了一项新指标，用于展示生成的查询从源表中正确检索到的值所占的比例。

为评估表检索性能，我们采用 recall@k (R@k)，即某个问题的真实表出现在其排名前 k 的检索结果中的比例，并对所有问题取平均；同时采用 perfect recall@k (PR@k)，即所有真实表均被检索到的问题所占的比例。在为该基准测试构建模型时，确定合适的 k 对端到端任务至关重要，这有助于在查询生成任务的开销（检索到的表越多，所需令牌也越多）与召回率之间取得平衡。

为评估查询生成，通常会使用生成的 SQL 查询以及执行查询后得到的输出。通常，精确匹配（EM）指标用于判断查询是否匹配。然而，随着查询变得更加复杂，执行后多个查询可能会产生相同或相似的输出。这凸显了基于执行的指标的重要性。因此，我们在 LOCuST 中采用以下指标。

1) 执行准确率（***EX***）和数值 ***F1*** 值（***NF1***）：EX 指标是文本到 SQL 中常用的指标，用于比较模型执行生成的查询所得的输出与

预期（标准）输出。这对于判断生成的查询与标准查询不匹配时的输出是否正确非常重要。对于该基准测试，计算 EX 时我们不考虑结果的排序。

EX 是一种二元指标。这意味着，当两个输出只有细微差异时，即使差异是由格式化造成的，EX 也会是零。它也无法捕捉部分正确的答案。此外，对于空结果集，EX 可能会因错误的原因而错误地判定模型表现良好。两个完全不同的查询可能会产生相同的空结果集。因此，我们在这种“严格”的 EX（EX_S）之外，还提供一种“宽松”变体（EX_L）。宽松 EX 会忽略结构差异（列名、形状、透视方向），仅比较真实值与预测输出中的单元格值多重集。此外，我们还报告数值 F1（NF1）指标，它是两个输出中数值观测值多重集之间的 F1 值。观测值 F1 指标（见下一段）进一步补充了这些基于输出的指标。

2) 度量、维度和观测值 ***F1***：为改进 EX 指标的二元特性，并衡量模型在回答问题时的接近程度，我们为度量、维度和观测值引入了一种新的 F1 指标。这些指标是强大的评估工具，可用于衡量模型在检索问题所询问的概念（*msr*F1）和筛选条件（*dim*F1）时，捕捉问题意图的能力。与精确匹配或执行准确率等二元指标相比，这些指标能提供更深入的见解。

度量 *F1*（*msrF1*）：度量 F1 用于确定语句选择度量的效果。度量 F1 通过解析 SQL 查询，并获取每个被引用表中的所有被引用列标识符来计算。接下来，取这些列标识符与知识图谱中连接到每个被引用表的所有度量的交集。由此得到查询 Q中引用的所有度量$\mathcal{M}$：

$$\mathcal{M}(Q) = C_Q \cap (\bigcup_{t \in \mathcal{T}_Q} \mathcal{M}(t, G))$$

其中，C_Q 是查询中引用的所有列标识符的集合，$\mathcal{T}_Q$ 表示查询中引用的所有表的集合，G 是语义图。我们根据预测查询和预期查询中同时被引用的度量来计算度量 F1。

维度 *F1*（维度*F1*）：维度 F1 用于衡量语句选择维度代码的效果。与*msr*F1 类似，可以通过解析 WHERE 和 PIVOT-FOR 语句中的列标识符和过滤字面量来计算维度 F1。SQL 中的列标识符与维度组相同，而过滤字面量则指维度代码（另请参见表 IIb 中的示例）。为了计算维度 F1，我们关注查询返回的所有维度代码。为此，我们需要从 SQL 中提取所有显式引用的维度代码（已过滤的代码），以及通过选择某个维度组但未对其进行过滤而返回的所有维度代码（未过滤的代码）。同样，对于被引用表，我们从解析后的 SQL 中提取所有列标识符和过滤字面量，并与图中每个被引用表的维度取交集。

首先，我们将查询 Q 的未过滤维度代码集合 $\mathcal{U}$ 定义为：

$$\mathcal{U}(Q) = \bigcup_{\substack{t \in \mathcal{T}_Q \\ g \in C_Q \cap \mathcal{D}_g(t,G) \\ L_Q \cap \mathcal{V}(g,G) = \emptyset}} \mathcal{V}(g, G)$$

其中，$g \in C_Q \cap \mathcal{D}_g(t,\ G)$ 表示查询中表 t 在语义图中的一个维度组。L_Q 表示查询中过滤子句中出现的所有字面值的集合，而 $\mathcal{V}(g,\ G)$ 表示图中与维度组 g 对应的维度代码集合。

其次，我们将已过滤维度代码集合 $\mathcal{F}$定义为：

$$\mathcal{F}(Q) = L_Q \cap (\bigcup_{t \in \mathcal{T}_Q} \mathcal{D}_c(t, G))$$

其中，$\mathcal{D}_c$ 表示图中某个表的所有维度代码集合。

最后，我们可以合并预测查询和真实查询中的未过滤与已过滤维度代码$\mathcal{D}_q(Q) = \mathcal{U}(Q) \cup \mathcal{F}(Q)$，并计算精确率和召回率，从而得出维度 F1。

观测值 *F1*（*obsF1*）：综合而言，*msr*F1 值和*dim*F1 值共同得出观测值 F1。观测值 F1 用于衡量语句选择正确观测值的效果。计算观测值 F1 时，会综合度量和维度的精确率与召回率，因为一个观测值由一个度量和一个或多个维度代码组成。如果预测查询与真实查询之间没有对应的度量，则*obs*F1 值为 0：

$$P_{\mathcal{O}}(Y, \hat{Y}) = \frac{\left|\mathcal{M}(Y) \cap \mathcal{M}(\hat{Y})\right| + \left|\mathcal{D}_q(Y) \cap \mathcal{D}_q(\hat{Y})\right|}{\left|\mathcal{M}(\hat{Y})\right| + \left|\mathcal{D}_q(\hat{Y})\right|}$$

$$R_{\mathcal{O}}(Y, \hat{Y}) = \frac{\left|\mathcal{M}(Y) \cap \mathcal{M}(\hat{Y})\right| + \left|\mathcal{D}_q(Y) \cap \mathcal{D}_q(\hat{Y})\right|}{|\mathcal{M}(Y)| + |\mathcal{D}_q(Y)|}$$

$${}_{obs}F1(Y, \hat{Y}) = \begin{cases} 0 & \text{if } \mathcal{M}(Y) \cap \mathcal{M}(\hat{Y}) = \emptyset \\ 2\frac{P_{\mathcal{O}} \cdot R_{\mathcal{O}}}{P_{\mathcal{O}} + R_{\mathcal{O}}} & \text{otherwise} \end{cases}$$

*obs*F1 值衡量正确检索出的观测值所占的比例，而不考虑查询输出是否进行了聚合或格式化。只要能够正确解析列标识符，即使 SQL 在语法上不正确，也可以计算 *obs*F1 值。因此，这些 F1 指标可以按“严格”方式计算（如果 SQL 不可执行，则设为 0），也可以按“宽松”方式计算（无论 SQL 中是否存在错误，始终计算 F1 分数）。

需要注意的是，当不区分表列中的度量和维度时（统计学以外的文本到 SQL 应用场景可能属于这种情况），仍然可以使用*dim*F1 指标：将所有表列视为维度，并用字面值过滤器L_Q 替代任何类型的列值过滤。我们建议未来文本到 SQL 研究在使用现有且常用的指标（如执行准确率）的同时，也采用该指标，以便更深入地了解模型检索预期值的能力，而不受对检索数据进行聚合或变换的影响。

IV. 实验设置

以下基线方法用于整体流程中的表检索和查询生成部分。

表检索： 向这些方法提供知识图谱中所有表的文本属性，包括表名、摘要和详细描述。随后，将每个表的这些文本属性拼接起来，以构建索引。作为第一个基线，我们提供一种基于 BM25+ [21] 的稀疏方法，并结合词干提取和停用词移除来创建简单的搜索索引。

其次，我们提供一种稠密方法，将每个表的所有文本属性转换为嵌入表示。利用训练数据集，我们根据对应答案，为每个问题提取正确的表和节点。随后，使用困难负例挖掘，为每对问题与表抽取 10 个样本表，以此训练一个用于表检索的小型 ColBERTv2 模型 [22] 。

查询生成： 基线方法是采用大语言模型和少样本提示策略。端到端任务的提示包含问题以及来自元数据知识图谱的表元数据，这些元数据基于使用我们的 ColBERT2 表检索基线检索到的前 5 个表及其对应节点。对于查询生成任务，每个问题仅提供相关表和知识图谱元数据。给定相关表后，我们使用 ColBERT2 模型为这些表检索带分数的节点，并在提示策略中使用这些分数。

我们通过为每个给定/检索到的表构建子图来序列化元数据图，并向大语言模型提示提供一个分层列表，其中包含每个表的所有度量组和维度组及其底层代码。对于每个节点，我们提供其标签、QUDT单位（适用于度量）、分区组信息（*xkos:isPartOf*，例如，市镇 X 属于省 Y）、总体性指示和相关性分数。对于包含时间和地理维度的表，我们会在提示中概述这些维度，以说明表中包含的内容，而不是冗长地列出这些维度组中数千个重复的特征。

每个提示都会说明如何针对不同的问题类型格式化 SQL 查询，以及如何对检索到的度量进行透视以获得答案。提示还会说明如何使用知识图谱中的元数据来确定要选择哪些度量和维度，并包含有关如何避免初期实验中观察到的常见错误的指导，以及两个端到端问答

表IV：基线表检索性能。

		@2		@5		@10	
	模型	**R**	**PR**	**R**	**PR**	**R**	**PR**
en	BM25+	0.25	0.22	0.36	0.33	0.44	0.40
	ColBERTv2	0.62	0.56	0.71	0.65	0.77	0.71
nl	BM25+	0.11	0.11	0.14	0.13	0.17	0.16
	ColBERTv2	0.40	0.36	0.61	0.56	0.75	0.71

示例，其中包含相关的图谱元数据，以及针对给定示例问题撰写的分析。[1]

使用了四个开箱即用的大语言模型：Claude5 Sonnet [23], GPT 6 Sol [24], LLama 4 Maverick [25] 和 Qwen 3.8- 27B [26]。

V. 实验分析

本节讨论基线模型在基准测试中的结果，并研究它们在各项任务中的表现。随后，我们将更详细地分析查询错误。首先，我们讨论基线模型在三项不同任务上的结果，并重点介绍枢轴消融中的关键结果。

A. 结果

1) 表检索： 表IV展示了两种基线表检索模型在两种语言上的 recall@k 分数。对于两种模型，在每个 k 下，英语结果始终高于荷兰语测试结果。BM25+ 的分数差异——BM25+ 是一种稀疏模型——表明，与英语相比，荷兰语测试集问题中使用的概念和行话与相应答案表之间存在更大的差异，如第 III-C 节所述。有趣的是，在荷兰语中，与 BM25+ 相比，ColBERTv2 稠密基线从更高的 k 中获益相对更多。这可能表明，表元数据中有大量信息并不能帮助区分表格。计算方法的描述以及许多包含相似、非描述性标准文本的表，可能会让稀疏模型更难区分彼此相似但存在细微差异的表，而这类表可能有很多。例如，区域统计数据中存在跨越多年的逐年独立表，因为市镇可能逐年发生变化。从上下文来看，这些表对于稀疏模型而言几乎完全相同，唯一具有区分度的特征可能只是表中提及的报告年份。这凸显了我们的数据集在正确进行表格消歧以回答问题方面所面临的挑战。

2) 查询生成： 表 V 展示了所有基线大语言模型在查询生成和端到端问答任务上的结果。在英语数据集的查询生成任务中，Claude 和 GPT 的总体得分最高，F1 值分别达到 0.84 和 0.87。荷兰语的得分也相近。对于未返回答案的情况（“不适用”列），模型出现退化，在达到最大输出词元数上限之前未能返回任何输出。即使在这些模型的上下文窗口中提供了正确的表，对于仅有 31%

表 V：查询生成 以及英语（en）和荷兰语（nl）数据集上所有基线模型的端到端结果 .

查询生成

	模型	不适用 执行准确率 s		EX_L	NF1	*msr*F1 值	*dim*F1 值	*obs*F1 值
en	Claude 5	0	0.12	0.16	0.64	0.89	0.90	0.84
	GPT 6	0	0.10	0.10	0.74	0.92	0.92	0.87
	LLama 4	3	0.03	0.06	0.50	0.82	0.80	0.73
	Qwen 3.8	0	0.09	0.11	0.56	0.82	0.81	0.75
nl	Claude 5	0	0.06	0.10	0.54	0.92	0.80	0.80
	GPT 6	0	0.07	0.10	0.64	0.95	0.84	0.84
	LLama 4	6	0.06	0.09	0.41	0.78	0.66	0.65
	Qwen 3.8	0	0.07	0.08	0.45	0.78	0.68	0.68

（a）查询生成任务中的性能。“NA”表示模型未能生成查询。

端到端问答

	模型	不适用 执行准确率 s		EX_L	NF1	*msr*F1 值	*dim*F1 值	*obs*F1 值
en	Claude 5	2	0.07	0.11	0.42	0.62	0.71	0.57
	GPT 6	0	0.10	0.11	0.52	0.66	0.77	0.62
	LLama 4	11	0.05	0.07	0.30	0.53	0.57	0.47
	Qwen 3.8	0	0.07	0.09	0.38	0.57	0.63	0.50
nl	Claude 5	2	0.03	0.07	0.31	0.63	0.60	0.54
	GPT 6	0	0.03	0.07	0.42	0.70	0.69	0.62
	LLama 4	13	0.03	0.05	0.19	0.47	0.43	0.38
	Qwen 3.8	1	0.05	0.08	0.30	0.58	0.56	0.49

（b）使用 ColBERTv2 方法作为表检索器时所有模型的端到端性能。

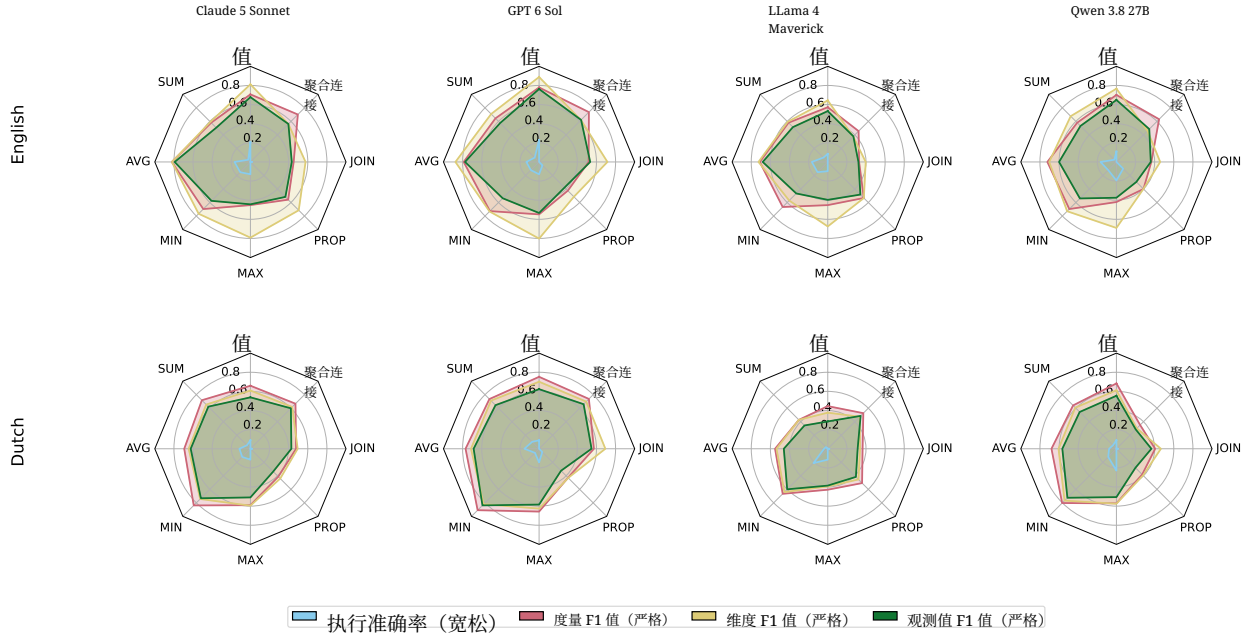

图 4：执行准确率以及 每个基线模型按 b 拆分的度量、维度和观测值 F1 分数
y 问题类型的端到端任务 ask.

（LLama）至 70%（GPT）的两种语言问题，生成的 SQL 中才能检索到所有正确的观测值（F1 值为 1）；而答案完全正确的比例最高也不超过 16%（Claude）。这些结果表明，即使在识别并选择与问题相关的列时，我们的数据集仍对所有模型的正确查询生成能力构成挑战。从 F1 值与执行准确率 $_L$ 之间的巨大差距中可以最清楚地看出这一点：F1 值是根据生成的 SQL 查询中所选和筛选的列计算得出的，不受特定问题类型所引入的聚合和格式化影响。

3) 端到端结果： 表 Vb 展示了使用基于表现最佳的 ColBERTv2 检索器的基线模型的端到端任务结果。查询生成与端到端任务之间的得分差异表明，在提示中提供正确的表并进行正确的模式链接，对于正确的查询生成非常重要。

与给定正确表来生成查询相比，在端到端任务中，所有模型最终的度量F1、维度F1 和观测值F1 都有所下降——

理论上，即使选择了与预期答案不同的表，度量、维度和观测值 F1 分数仍可能表明，模型查询的是另一个表，但使用了相似的度量和/或维度。然而，正如结果所示，实际情况往往并非如此。

*4)*执行准确率： 在查询生成任务中，所有模型的严格 EX 介于 0.03 到 0.12 之间，宽松 EX 介于 0.06 到 0.16 之间。在端到端任务中，除英语任务上的 GPT 和 LLama 外，所有模型的得分都更低。EX 与观测值F1 值之间的差距表明，大多数度量和维度代码都选对了，但遗漏或多选少量代码，就会导致基线模型无法输出预期结果。较低的 EX 不仅反映了答案错误，也反映了正确答案在格式上的差异。在查询生成任务中，每次运行有 17–53% 的问题是通过查询作答的，该查询选出的观测值与预期完全一致，返回的

预期值（观测值F1 值和 NF1 均为 1），但宽松 EX 得分为 0。在这些情况下，结果通常仅在列方面与预期答案不同。例如，对于问题‘阿姆斯特尔芬文化、体育和娱乐行业在 *2010* 年交付了多少电量？’，GPT 6 仅返回所请求的值，而预期结果还包含筛选维度的列（地区、所有时期和行业分支SIC2008）。可以说，标准 SQL 和生成的 SQL 都是正确的；它们检索到了正确的观测值。因此，基于 EX 的指标取决于结果约定，而自然语言问题往往会省略这些约定；观测值F1 值则不受其影响，因为它只考虑查询选取了哪些数据。

5) 各问题类型的结果：图4展示了所有模型按问题类型细分的端到端性能。在英语数据集上取平均后，就 *obs*F1 而言，模型在 AVG-（0.71）和 VALUE 类（0.66）问题上的得分最高。JOIN-（0.41）、PROP-（0.42）和 MAX 类（0.44）查询的表现最弱。SUM 类和 MIN 类问题的模型平均 *obs*F1 为 0.53，而 AGGJOIN 为 0.51。这与我们在第三节中推测的问题类型难度大体一致。AGGJOIN 类问题在 *obs*F1 方面是个例外，但在难度方面并非如此。所有这些问题都将具有相同度量的两个表结合起来，因此，即使只查询其中一个表，也仍能选出大多数预期代码，从而抬高 *obs*F1 得分。AGGJOIN 答案中只有 9% 的 *obs*F1 为 1，是所有问题类型中最低的，而且没有一个答案完全正确。JOIN 类和 AGGJOIN 类查询的得分还凸显了正确进行表和模式链接对于生成正确 SQL 的重要性。在仅查询生成任务与端到端任务之间，JOIN 类查询的模型平均 *obs*F1 从 0.82 降至 0.41；在语法有效的端到端查询中，仅有 27%（JOIN）和 18%（AGGJOIN）使用了正确的表。PROP 类查询本身似乎就很难，因为它们在两项任务中的得分都较低（查询生成任务为 0.63，而端到端任务为 0.42），尽管语法有效的端到端 PROP 查询中有 75% 使用了正确的表。

B. 分析

*1)*观测值***F1*** 值弥补了执行准确率的不足：表 VI 展示了所有模型在查询生成和端到端任务上计算出的所有指标的斯皮尔曼相关矩阵。在常规 SQL 运行中，对于所有带有生成查询的问题（$n = 4$，842 个），严格 EX 和宽松EX 在 94% 和 91% 的查询对中均为零，因此对于绝大多数生成的查询而言，无法获得有用的统计结果。相比之下，观测值F1 值具有梯度，并且对于宽松EX 为 0 的查询，其中 80% 可以获得部分得分。它与严格 EX（$\rho = 0.22$）和宽松 EX（0.24）的相关性较弱，表明它捕捉的是不同于执行正确性的方面，即无论聚合和格式化如何，是否选定了正确的观测值。在基于输出的指标中，它与 NF1 的相关性最高（0.63），表明选定正确的观测值通常也意味着返回正确的值。观测值F1 值与其两个组成部分均具有相关性（与度量F1 值为 0.72，与 维度F1 值为 0.81），而这两个组成部分彼此之间的相关性较弱（0.34）。因此，观测值F1 值并不冗余于维度F1 值：只有 48% 的 SQL 中二者相等，另有 19% 的 SQL 中二者相差超过 0.1。这些差异主要源于查询未选定任何预期的度量；对于这类查询，观测值F1 值为零，而维度F1 值的平均值为 0.51。相反，它很少会惩罚正确答案。生成的 SQL 中，只有 1.3% 的查询宽松 EX 为 1 且观测值F1 值低于 1。最后，在荷兰语中，我们的基线模型之间的 EX 最多相差 5 个问题。这无法可靠地区分各模型，而它们的观测值F1 值和 NF1 分别介于 0.65 至 0.84 和 0.41 至 0.64 之间。观测值F1 值无法评估应用于选定观测值的正确聚合和范围筛选器，而 EX 和 NF1 可以评估。观测值F1 值为 1 的可执行查询中，只有 76% 的查询 NF1 也为 1。因此，观测值F1 值是对基于输出的指标的补充，而非替代。

表 VI：查询生成和端到端任务中（宽松）评估指标之间的斯皮尔曼相关系数，针对生成了查询的所有问题。*

指标	$\mathbf{EX_S}$	$\mathbf{EX_L}$	**NF1**	***msr*F1 值**	***dim*F1 值**	***obs*F1 值**
$\mathbf{EX_S}$		0.83	0.31	0.15	0.18	0.22
$\mathbf{EX_L}$	0.83		0.37	0.17	0.20	0.24
NF1	0.31	0.37		0.43	0.54	0.63
***msr*F1 值**	0.15	0.17	0.43		0.34	0.72
***dim*F1 值**	0.18	0.20	0.54	0.34		0.81
***obs*F1 值**	0.22	0.24	0.63	0.72	0.81	

* 4 模型、英语 和荷兰语、查询生成和端到端任务；$n = 4$，842

2) 语法错误主要是绑定错误：我们发现了一个主要的语法错误来源。在所有模型和两种语言中，2,431 个生成的查询中有 158 个（6.5%）无法执行。其中几乎全部（148 个）是绑定错误，而不是解析错误。最常见的情况（123 个）是查询引用了在使用位置不可用的列。例如，在一个未选择 Periods 的 UNPIVOT 子查询之后按 Periods 进行筛选，或者使用了拼写错误或虚构的列名，或将维度代码用作列名。

*3)*未查询所有要求的度量和维度是最常见的错误：表八展示了我们在分析生成的 SQL 查询时发现的一些常见具体问题。大多数错误源于生成的查询未引用预期的度量筛选条件和维度筛选条件（“缺失”），或查询了预期答案中没有的度量或维度（“多余”，通常是为了替代某个缺失的筛选条件）。在查询生成任务中，因模型而异，查询在 18–23%（英语）和 27–35%（荷兰语）的情况下缺少预期的维度筛选条件；在 14–36% 和 26–52% 的情况下，查询会针对非预期维度进行筛选。相比之下，缺失或多余度量的比例为 3–16%。在缺少预期维度的查询中，有 60–75% 的查询里，模型还会筛选错误的代码，而非仅仅省略该筛选条件。此外，模型会在相同的问题上犯这些错误。在所有模型都作答的 307 个英语问题和 294 个荷兰语问题中，有 45 个英语问题和 79 个荷兰语问题上，四个模型都

表 VII：英语测试集上所有模型在端到端问答任务中，各问题类型的查询数量、语法错误和错误选择的聚合，以及测试问题中各域的语法错误数量。语法错误和百分比均以生成的查询数量为基准；错误聚合则以不含语法错误的生成查询数量为基准。

		Claude 5		GPT 6		LLama 4		Qwen 3.8	
Type	**# 问题数量**	**语法错误**	**错误聚合**	**语法错误**	**错误聚合**	**语法错误**	**错误聚合**	**语法错误**	**错误聚合**
值	83	4（5%）	8（10%）	0（0%）	0（0%）	16（20%）	22（35%）	7（8%）	15（20%）
SUM	31	4（13%）	3（12%）	1（3%）	4（13%）	4（13%）	2（8%）	1（3%）	1（3%）
AVG	30	0（0%）	2（7%）	0（0%）	1（3%）	3（10%）	0（0%）	1（3%）	0（0%）
MIN	33	1（3%）	1（3%）	0（0%）	1 (3%)	4 (13%)	0 (0%)	1 (3%)	0 (0%)
MAX	31	0 (0%)	6（19%）	0（0%）	4（13%）	1（4%）	0（0%）	3（10%）	1（4%）
JOIN	63	6（10%）	38（67%）	2（3%）	37（61%）	22（36%）	26（67%）	22（35%）	26（63%）
PROP	20	1（5%）	6（33%）	7（35%）	1（8%）	5（25%）	1（7%）	6（30%）	1（7%）
聚合连接	19	3（16%）	12（75%）	3（16%）	13（81%）	9（47%）	6（60%）	6（32%）	10（77%）

		语法错误			
域	**问题数**	**Claude 5**	**GPT 6**	**LLama 4**	**Qwen 3.8**
教育	54	1（2%）	2（4%）	9（18%）	3（6%）
能源	121	8（7%）	4（3%）	25（22%）	18（15%）
国际贸易	20	3（15%）	0（0%）	6（30%）	6（30%）
制造业	66	7（11%）	5（8%）	17（27%）	14（21%）
自然与环境	5	0（0%）	0（0%）	0（0%）	0（0%）
安全与司法	38	0（0%）	2（5%）	5（13%）	5（13%）

表八：英语测试集上各模型在查询生成任务中的常见错误，以已生成查询的百分比表示（“未生成查询”则以所有问题的百分比表示）。

错误类型	**Claude 5**	**GPT 6**	**LLama 4**	**Qwen 3.8**
未生成查询	0.0	0.0	1.0	0.0
语法错误	2.6	1.0	6.5	7.7
错误的表*	4.3	3.6	4.2	5.9
错误聚合*	8.6	4.6	19.9	11.5
缺失度量	10.6	8.4	15.6	13.9
额外度量	10.6	9.0	13.4	13.2
缺失维度	19.0	17.7	22.8	21.3
额外维度	20.6	14.2	36.2	26.8

*生成的查询中不含语法错误的百分比。

选择了错误的维度。因此，瓶颈在于依据问题及其表格确定正确的维度代码，而非某个特定模型。错误表源于在 JOIN 类和 AGGJOIN 类查询中只使用了给定表中的一张，尽管查询生成任务中已提供所有正确的表。

C. 消融实验：透视

我们将数据集中的表建模为数据立方体。由于数据具有多维特性，在官方统计中，无论是较小的“展示”表还是较大的“参考”表，使用透视表都是一种常见做法 [27]。这也意味着，在统计学中，使用透视查询是标准做法，可返回这些多维数据的切片。查询数据时返回透视的“展示”表，有助于生成易读的二维表格结果。因此，我们开展了一项消融实验，预期模型会根据问题中的指示，对查询所检索的所有度量或指定维度返回透视结果。示例2和表 IX 展示了一个说明这一现象的答案示例。如由于透视仅仅是对 SQL 输出进行转换，透视查询与非透视查询返回的信息（即观测值）相同。

表 IX：示例2按度量进行透视后的查询结果（橙色）。颜色表示维度代码（浅绿色）和观测值（蓝色）。标签和单位取自语义图。

度量	**Unit**	**少于101名学生** **罗马天主教小学** **平均值['时期']**
在籍学生	人数	13755.0

表X比较了在英语和荷兰语查询生成任务中，透视与常规SQL生成的表现。如果考虑宽松的*obs*F1 值（计算时也纳入不可执行查询，类似于宽松EX），那么无论对哪种模型和语言，透视与常规SQL生成之间的差异都不超过0.04。这表明，透视并不会改变模型选择的数据。不过，所有模型的严格分数均有所下降（*obs*F1 -0.08 至 -0.22），因为含有语法错误的查询占比在英语中从1-8%上升至11-34%，在荷兰语中从3-16%上升至18-43%。较弱的LLama模型受影响最大，其*obs*F1 值在英语和荷兰语中分别下降了0.20和0.22。

有趣的是，所有模型的宽松 EX 都有所提升（荷兰语上的 LLama 除外）。这并非因为查询质量提高，而是输出形式所致：预期的常规 SQL 输出也包含已过滤维度代码的列（见表 IIb），而模型似乎经常会遗漏这些列。另一方面，透视查询会将这些维度移至列标题中（见表 IX），而计算宽松 EX 时会忽略这些列标题（见 III-D1 节）。在 NF1 为 1 的可执行查询中（1,141 条常规查询和 720 条

条透视查询），常规 SQL 查询中有 15%–31% 的宽松 EX 得分为 1，而透视 SQL 查询中这一比例为 56%–84%。生成透视查询确实显著增加了语法错误的数量（从所有模型合计 158 个增至 651 个）。其中一半语法错误（323 个）是由无效的 PIVOT 语法导致的解析错误。当模型改为生成常规 SQL 时，解析错误降至仅 10 个，而绑定错误则减少了一半以上（从 328 个降至 148 个）。因此，透视给模型增加了它们难以完成的输出转换任务。不过，透视并未实质性影响模型的依据。

```
SELECT * FROM (
  SELECT 度量, 值, 时期, 教育类型,
    意识形态基础, 学校规模
  来自 "data/英语/odata3/03753eng.parquet"
  UNPIVOT (值 FOR 度量 IN (
    在校学生人数Students_2'))
  其中 时期 属于 ('2004SJ00',          '2003年SI00')
)
透视(平均值 (值) 按
  教育类型 属于 ('A025280')
  意识形态基础 属于 ('A041917')
  学校规模属于 ('A041906')
  分组依据 度量)
```

示例2：针对问题*‘2003*年和*2004*年期间，小型罗马天主教小学的平均在校学生人数是多少？’生成的透视 SQL；该查询与其未透视版本查询相同的观测值（见示例1）。

表X：各模型在带有透视要求的纯查询生成任务上的表现。常规SQL生成的表现有所提升或下降，分别以 +和-表示。

	模型	EX_S	EX_L	*msr*F1 值	*dim*F1 值	观测F1 值
en	Claude 5	$0.07_{-0.05}$	$0.31_{+0.15}$	$0.81_{-0.08}$	$0.81_{-0.09}$	$0.76_{-0.08}$
	GPT 6	$0.08_{-0.02}$	$0.38_{+0.28}$	$0.75_{-0.17}$	$0.76_{-0.16}$	$0.72_{-0.15}$
	LLama 4	0.03	$0.13_{+0.07}$	$0.57_{-0.25}$	$0.56_{-0.24}$	$0.53_{-0.20}$
	Qwen 3.8	0.09	$0.25_{+0.14}$	$0.64_{-0.18}$	$0.64_{-0.17}$	$0.61_{-0.14}$
nl	Claude 5	$0.05_{-0.01}$	$0.21_{+0.11}$	$0.77_{-0.15}$	$0.68_{-0.12}$	$0.68_{-0.12}$
	GPT 6	$0.04_{-0.03}$	$0.27_{+0.17}$	$0.76_{-0.19}$	$0.68_{-0.16}$	$0.67_{-0.17}$
	LLama 4	$0.00_{-0.06}$	$0.06_{-0.03}$	$0.51_{-0.27}$	$0.44_{-0.22}$	$0.43_{-0.22}$
	Qwen 3.8	$0.03_{-0.04}$	$0.13_{+0.05}$	$0.54_{-0.24}$	$0.46_{-0.22}$	$0.46_{-0.22}$

VI. 讨论与未来工作

半自动标注与人工标注之间可能存在质量差异： 正如第三节 C 部分所述，与英语数据集相比，荷兰语数据集的标注过程需要更多人工标注，这可能影响了模型在基准测试任务中的结果。然而，这确实凸显了人工标注对于现实世界文本到 SQL 基准测试的重要性，尤其是对于荷兰语这类低资源语言而言，因为人工标注有助于生成更加富有创意且具有歧义的问题表述。正如标注人员所指出的，机器生成的英语问题往往更加冗长，会详细罗列答案表中的每一项信息。因此，尽管这些问题忠实反映了答案，但其问题表述的创意性和歧义性低于现实世界用户可能采用的程度。

应当研究无法回答的问题： 测试集中没有包含要求对不兼容计量单位进行聚合的问题，尽管语义知识图谱支持此类操作。其他类型的“无法回答”问题也是如此。这类问题可能出于各种原因，例如语义错误问题、数据不完整性、歧义、包含错误前提的问题（例如，“荷兰哪些城市的居民人数多于阿姆斯特丹？”），或随着时间推移，针对表面上相同但计算方法不同的度量进行聚合。未来的工作应研究这些类型的问题，并将其纳入训练集和测试集。这一点非常重要，因为现实世界场景中会遇到这些无法回答的问题，尤其是在提供事实信息至关重要的环境中。向用户说明哪些数据不可用、已撤回或不应披露，以及其中原因，可能会提供极具价值的信息。

VII. 结论

本文介绍了 LOCuST 基准及其数据集，这是一项用于真实世界英语和荷兰语统计数据的文本到 SQL 自然语言问答资源。现有数据集通常包含无法体现真实世界官方统计复杂性的数据，而且往往未在开放域目录规模的端到端检索和查询生成任务上进行测试。

我们提供了针对荷兰统计局提供的 2,241 个真实世界开放数据表的专家标注问答对，以及一个包含表格元数据（如描述、标签和计量单位）的语义知识图谱。我们引入观测值 F1 指标，用于确定给定答案中有多少部分相对于预期答案是正确的，即使生成的 SQL 查询与预期的 SQL 查询之间存在差异。

最先进的大语言模型基线的开箱即用结果表明，当前模型在为统计数据问题生成正确 SQL 方面仍面临困难。该基准反映了统计数据最终用户真实且常见的信息需求。考虑到这一点，模型在生成查询时，即使编写常规 SQL，或在官方统计中常见的需要对输出结果进行透视时，仍然无法在这些数据上取得理想表现。因此，该基准测试了模型当前能力与该域现实应用之间的实际差距，并表明未来需要针对大规模、复杂现实场景中的文本到 SQL 展开研究；在考虑目录规模的文本转 SQL 时，也需要更智能的系统。此外，我们的基准还为荷兰语这一低资源语言提供了测试平台。我们希望 LOCuST 基准能为研究文本到 SQL 系统的实践者和研究人员提供有价值的资源。

AI生成内容声明

本研究使用生成式人工智能工具的方式如下。Claude Code（Anthropic）协助测试和验证基准测试代码，包括审查 S 表达式到 SQL 的转换、评估脚本、从 OData3 API 填充元数据图谱的代码以及数据处理工具。所有代码均由作者审查并测试。Claude Code 还用于对问题—查询对进行数据质量审计，以标记可能存在歧义或不匹配的条目，并提出修复方案。每项提议的修改均由人工审查并批准。Claude 用于语言编辑，并协助完成本文的 LaTeX 格式化。所有科学论断、分析和结论均为作者本人提出，作者对本文内容承担全部责任。

参考文献

[1] A. Bo˘zi˘c Verbi˘c，“官方统计与政府决策：政策相关学术研究的文献计量与主题分析”，《官方统计学杂志》，第 41 卷，第 4 期，第 984–1020 页，2025 年。[在线]。可访问：https://doi.org/10.1177/0282423X251339860[2] 《欧洲商业统计手册》——*2021* 年版，Eurostat，卢森堡，2021 年。 [在线]。可访问：https://ec.europa.eu/eurostat/web/products-manuals-and-guidelines/-/ks-gq-21-001[3] 欧洲议会和欧盟理事会，“欧洲议会和理事会于 2009 年 3 月 11 日通过的关于欧洲统计的法规（EC）第 223/2009 号”，《欧盟官方公报》，L 87，第 164–173 页，2009 年 3 月，2024 年 12 月 26 日合并版本。 [在线]。可访问：https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=CELEX%3A02009R0223-20241226[4] L. Shi、Z. Tang、N. Zhang、X. Zhang 和 Z. Yang，“关于采用大语言模型执行文本到SQL任务的综述”，《*ACM*计算调查》，第 58 卷，第 2 期，第 54:1–54:37 页，2025 年。 [在线]。可访问：https://doi.org/10.1145/3737873 。/ O. Bouattour、J. Weiner、N. Tatbul、F. Wenz、P. B. Chen、L. Mills、Y. He、B. Arbab、C。Demiralp 和 M. Stonebraker，“ 企业级文本到 SQL 的提示策略：一项消融研究 ”，载于 *VLDB 2026* 年第 *2* 届新型视觉人工智能系统优化研讨会，2026 年。 [在线]。可访问：https://vldb.org/2026/Workshops/VLDB-Workshops-2026/NOVAS/NOVAS26 16.pdf[6] T. Yu、R. Zhang、K. Yang、M. Yasunaga、D. Wang、Z. Li、J. Ma、I. Li、Q. Yao、S. Roman、Z. Zhang 和 D. Radev，“Spider：用于复杂及跨域语义解析和文本到SQL任务的大规模人工标注数据集”，载于*2018* 年自然语言处理经验方法会议论文集。比利时布鲁塞尔：计算语言学协会，2018 年 10 月至 11 月，第 3911–3921 页。 [在线]。可访问：https://aclanthology.org/D18-1425/[7] F. Lei、J. Chen、Y. Ye、R. Cao、D. Shin、H. Su、Z. Suo、H. Gao、W. Hu、P. Yin、V. Zhong、C. Xiong、R. Sun、Q. Liu、S. Wang 和 T. Yu，“Spider 2.0：评估语言模型在真实世界企业级文本到SQL工作流中的表现”，载于国际学习表征会议（*ICLR*），2025 年。[在线]。可访问：https://proceedings.iclr.cc/paper files/paper/2025/hash/46c10f6c8ea5aa6f267bcdabcb123f97-Abstract-Conference.html[8] J. Li、B. Hui、G. Qu、J. Yang、B. Li、B. Li、B. Wang、B. Qin、R. Geng、N. Huo、X. Zhou、C. Ma、G. Li、K. C.-C. Chang、F. Huang、R. Cheng 和 Y. Li，“LLM 是否已经能够充当数据库接口？面向大规模数据库增强文本到SQL查询的 BIg 基准测试”，载于《神经信息处理系统进展》第 *36* 卷（*NeurIPS 2023*）2023 年。 [在线]。可访问：https://papers.nips.cc/paper files/paper/2023/hash/83fc8fab1710363050bbd1d4b8cc0021-Abstract-Datasets and Benchmarks.html[9] C. Zhang、Y. Mao、Y. Fan、Y. Mi、Y. Gao、L. Chen、D. Lou 和 J. Lin，“FinSQL：面向金融分析的模型无关大语言模型文本到SQL框架”，载于*2024* 年国际数据管理会议附属会议，丛书：SIGMOD ’24。美国纽约州纽约：计算机协会，2024 年，第 93–105 页。 [在线]。可访问：https://doi.org/10.1145/3626246.3653375

[10] P. B. Chen、D. Yang、W. Li、F. Wenz、Y. Zhang、N. Tatbul、M. Cafarella、C。Demiralp 和 M. Stonebraker，“BEAVER：企业级文本到SQL基准测试”，2026 年，第 3 版，修订于 2026 年 5 月 13 日。 [在线]。可访问：https://arxiv.org/abs/2409.02038 。v F. Nooralahzadeh、Y. Zhang、E. Smith、S. Maennel、C. Matthey-Doret、R. De Fondeville 和 K. Stockinger，“StatBot.Swiss：以自然语言进行双语开放数据探索”，载于《计算语言学协会论文集》：ACL 2024 论文集，L.-W. Ku、A. Martins 和 V. Srikumar 编。泰国曼谷：计算语言学协会，2024 年 8 月，第 5486–5507 页。 [在线]。可访问：https://aclanthology.org/2024.findings-acl.326/[12] C. Deng、J. Fan、Y. Luo、Q. Xue、M. Fan、Y. Zhang、M. Zhang、X. Jia、J. Zhang 和 X. Du，“TACO：面向具有歧义性的开放域文本到SQL及跨数据库查询的基准测试”，《VLDB 论文集》，第 19 卷，第 6 期，第 1386–1399 页，2026 年 2 月。 [在线]。可访问：https://doi.org/10.14778/3797919.3797942 。/ . B. Li、Y. Luo、C. Chai、G. Li 和 N. Tang，“自然语言到 SQL 的黎明：我们是否已经完全准备就绪？ ”，《VLDB 论文集》，第 17 卷，第 11 期，第 3318–3331 页，2024 年 7 月。 [在线]。可访问：https://doi.org/10.14778/3681954.3682003[14]“《荷兰中央统计局法》”，2025 年，2003 年 11 月 20 日法案，BWBR0015926；截至 2025 年 9 月 1 日的合并版本。 [在线]。可访问：https://wetten.overheid.nl/BWBR0015926/2025-09-01[15] R. Cyganiak 和 D. Reynolds，“RDF 数据立方体词汇表”，万维网联盟（W3C），W3C 建议，2014 年 1 月。 [在线]。可访问：https://www.w3.org/TR/vocab-data-cube/[16] A. Miles 和 S. Bechhofer，“SKOS 简单知识组织系统参考”，万维网联盟（W3C），W3C 建议，2009 年 8 月。 [在线]。可访问：https://www.w3.org/TR/skos-reference/[17]R. Albertoni、D. Browning、S. J. D. Cox、A. Gonzalez Beltran、A. Perego 和 P. Winstanley，“Data Catalog 词汇表（DCAT）——第 3 版”，万维网联盟（W3C），W3C 建议，2024 年 8 月。 [在线]。可访问：https://www.w3.org/TR/vocab-dcat-3/[18] DCMI 使用委员会，“DCMI 元数据术语”，都柏林核心元数据倡议，DCMI 建议，2020 年 1 月。 [在线]。可访问：https://www.dublincore.org/specifications/dublin-core/dcmi-terms/[19] QUDT.org，“QUDT 本体”，访问日期：2026 年 11 月 11 日。 [在线]。可访问：https://www.qudt.org/[20] Y. Gu、S. Kase、M. T. Vanni、B. M. Sadler、P. Liang、X. Yan 和 Y. Su，“超越独立同分布：知识库问答的三个泛化层次”，载于《2021 年万维网大会论文集》，丛书：WWW ’21。美国纽约州纽约：计算机协会，2021 年，第 3477–3488 页。 [在线]。可访问：https://doi.org/10.1145/3442381.3449992 。/ . Y. Lv 和 C. Zhai，“词频归一化的下界”，载于《第 20 届 ACM 信息与知识管理国际会议论文集》，丛书：CIKM ’11。美国纽约州纽约：计算机协会，2011 年，第 7–16 页。 [在线]。可访问：https://doi.org/10.1145/2063576.2063584 。/ . K. Santhanam、O. Khattab、J. Saad-Falcon、C. Potts 和 M. Zaharia，“ColBERTv2：通过轻量级晚期交互实现高效检索”，载于《2022 年北美计算语言学协会会议：人类语言技术论文集》。美国西雅图：计算语言学协会，2022 年 7 月，第 3715–3734 页。 [在线]。可访问：https://aclanthology.org/2022.naacl-main.272/[23] Anthropic，“系统卡：Claude Sonnet 5”，Anthropic，技术报告，2026 年 6 月。 [在线]。可访问：https://www.anthropic.com/claude-sonnet-5-system-card[24] OpenAI，“GPT-6 Astra 系统卡”，OpenAI，技术报告，2026 年 9 月。[在线]。可访问：https://deploymentsafety.openai.com/gpt-6-astra/gpt-6-astra.pdf[25] Meta AI，“Llama 4 系列：原生多模态人工智能创新新时代的开端”，Meta AI 博客，2025 年 4 月。 [在线]。可访问：https://ai.meta.com/blog/llama-4-multimodal-intelligence/[26] Qwen 团队，“Qwen3.8-27B”，Hugging Face 模型卡片，2026 年 8 月。[在线]。可访问：https://huggingface.co/Qwen/Qwen3.8-27B[27] 联合国欧洲经济委员会，《让数据具有意义》第二部分：统计数据呈现指南。日内瓦：联合国，2009 年。 [在线]。可访问：https://unece.org/DAM/stats/documents/writing/MDM Part2 English.pdf