借助生成式人工智能自动化训练学习型数据库组件

Angjela Davitkova RPTU Kaiserslautern-Landau 凯撒斯劳滕，德国
angjela.davitkova@cs.rptu.de

Sebastian Michel RPTU Kaiserslautern-Landau 凯撒斯劳滕，德国
sebastian.michel@cs.rptu.de

arXiv:2512.20271v1 [cs.DB] 23 Dec 2025

摘要

深度学习在数据库优化中的应用日益广泛，为索引、基数估计和查询优化带来了改进。然而，获取高质量训练数据仍然是一项重大挑战。本文探讨了使用生成模型（如 GPT）为学习型数据库组件合成训练数据的可能性。我们开展了一项初步可行性研究，考察这些模型为数据库工作负载生成逼真的查询分布和执行计划的能力。此外，我们还讨论了数据可扩展性和标注等关键挑战，以及潜在的解决方案。初步结果表明，生成模型能够有效扩充训练数据集，从而提升学习型数据库技术的适应性。

ACM 引用格式：
Angjela Davitkova 和 Sebastian Michel。2025年。《借助生成式人工智能自动化训练学习型数据库组件》。载于前瞻性人工智能系统的新型优化（*NOVAS '25*），*2025*年*6*月*22*日至*27*日，德国柏林。ACM，美国纽约州纽约市，共5页。https://doi.org/10.1145/3735079.3735322

1 引言

现代数据库系统中的数据量和复杂性不断增加，因此需要持续优化索引、查询处理和基数估计。传统的数据库优化方法依赖人工设计的启发式方法和基于规则的技术，在面对不断变化的工作负载和大规模数据集时往往难以应对。近期机器学习（ML）的进展引入了学习型模型，可通过预测数据分布来改进数据库性能，用于基数估计 [12, 17, 22], 优化查询执行计划 [16, 20],以及增强索引机制 [4, 13]。尽管学习型数据库组件前景广阔，尤其是查询模型，但要使其有效运行，需要大量标注训练数据，因为它们依赖学习来识别模式并泛化到不同的数据集。然而，获取此类数据面临重大挑战。

首先，存在**真实世界数据不足**的问题。由于隐私问题以及难以访问生产数据库日志，包含键分布、查询工作负载和执行统计信息的高质量训练数据集难以获取。此外，学习型模型需要通过查询进行训练，因此面临**冷启动问题**：在缺乏足够初始训练数据的情况下，它们难以做出准确预测，导致性能不佳，直到获得足够有代表性的样本为止。

其次，在**数据偏差和分布偏移**存在的情况下，会出现问题。在静态数据集上进行训练通常会导致过拟合：模型学习到训练数据特有的模式，却无法泛化到未见过且不断演变的工作负载。这一局限性降低了模型的适应能力，使其在面对现实世界中的变化时表现欠佳，例如工作负载偏移、季节性趋势或动态变化的数据库内容。

第三，**可扩展性问题**也是一个额外的难题。手动生成和标注大规模数据集，或通过查询执行来生成和标注，计算成本极高，需要大量的处理资源和时间。在动态数据库环境中，这一挑战更加突出，因为数据分布和查询工作负载都在持续变化，因此需要频繁地重新生成和标注训练数据。这种开销使其难以应用于实际场景，尤其是在需要为索引、基数估计和查询优化维护准确且最新的学习型模型时。

为克服这些挑战，本文探讨了利用生成模型自动化并改进学习型数据库组件训练数据生成流程的可行性，重点关注基数估计和查询优化。我们着重研究生成模型生成多样且具有代表性的工作负载的能力，并识别与合成数据生成相关的关键因素。我们表明，GPT 等模型能够利用数据语义生成有意义的工作负载，为基于规则的查询生成提供了一种有前景的替代方案。此外，我们还讨论了模型的局限性，尤其是在处理选择率相关任务时，并提出了未来可能的研究方向，以提升 GPT 的适应性和稳健性，最终助力制定更高效、准确的数据库优化策略。数据生成还为测试和基准测试提供了途径，有助于设计更稳健的组件。

我们将在第 2 节介绍基数估计和查询优化的工作负载合成需求，并在第 3 节通过一项初步可行性研究探讨 GPT 在这方面的潜力，随后在第 4 节讨论相关工作，并在第 5 节简要总结并展望未来。

2 方法论

2.1 基数估计

深度学习方法在预测查询选择率方面展现出良好的前景，能够据此选择高效的执行计划来

特此授予免费许可，允许为个人或课堂使用而对本作品的全部或部分内容制作数字或纸质副本，前提是不得出于营利或商业利益而制作或分发副本，并且副本首页须保留本声明及完整引文。对于本作品中由非作者拥有版权的组件，必须尊重其版权。经注明出处后，允许制作摘要。若要以其他方式复制、重新出版、发布到服务器或重新分发至列表，必须事先获得具体许可和/或支付费用。请通过 permissions@acm.org 申请许可。*NOVAS '25*，德国柏林© 2025年版权所有归所有者/作者所有。出版权授权给 ACM。ACM ISBN 979-8-4007-1917-2/2025年/06 https://doi.org/10.1145/3735079.3735322

提升性能。然而，这些模型需要高质量的带标签数据集来进行训练、测试和持续改进。这项任务的复杂性在于，既要确保查询具有实际意义，又要考虑各种查询模式、数据分布和谓词选择率。

2.1.1 基数估计问题表述。 查询 Q 在数据库 D 中的选择率 $\sigma(Q, D)$，若元组总数为 $|D|$，则可估计为：$\sigma(Q, D) = \frac{|R(Q, D)|}{|D|}$ 其中，$R(Q, D)$ 是满足查询 Q 的元组集合。

2.1.2 训练数据生成。 为基数估计生成合成训练数据，关键在于创建多样化的查询工作负载并为其分配相应的选择率。**步骤 1：生成多样化的 SQL 查询** 训练基数估计器的第一步是生成涵盖不同类型工作负载且复杂度各异的查询。根据所测试的基数估计器，生成的查询可能涉及各种操作，从而产生不同结果：

- 简单查询：包含直接选择条件和投影的查询。
- 复杂连接：涉及多个连接的查询，需要估算中间结果的基数。
- 聚合：涉及分组子句的查询，由于数据分组，需要进行复杂的基数估计。

步骤 2：生成准确标签 包括为生成的查询指定选择率或实际基数值。这些值通常来自对数据执行实际查询，或通过采样技术进行近似估算。尽管 GPT 能够根据从数据中学习到的模式生成查询语法和结构，但它无法实时分析底层数据分布。我们将在可行性研究（第 3 节）中讨论模型能力的范围。

2.2 查询优化

随着学习型查询优化器在处理复杂且动态的工作负载方面日益受到关注，生成高质量的训练数据对于有效训练这些优化器至关重要。

2.2.1 查询优化的问题表述。 给定一个查询Q和一个数据库 D，目标是选择一个候选执行计划 P，使执行查询 Q的代价 $C(Q, P, D)$ 最小，即 $P_{\text{opt}}(Q, D) = \arg\min_P C(Q, P, D)$。

2.2.2 训练数据生成。 为查询优化生成合成训练数据的过程可以分解为生成多样化的 SQL 查询、为这些查询模拟多个执行计划，以及估算代价。

步骤 1：生成多样化的 SQL 查询 对于训练查询优化器至关重要。与基数估计不同，这一步通常需要涉及多个连接和嵌套查询的更复杂查询。

步骤 2：查询-计划对生成包括为每个生成的 SQL 查询 Q 模拟多个候选执行计划（P_1、P_2..... P_n）。这些执行计划代表执行查询的不同策略，例如：

- 连接顺序：表之间连接的不同顺序。
- 连接方法：哈希连接、嵌套循环连接和归并连接。
- 访问方法：索引扫描、全表扫描。

为单个查询模拟多个候选执行计划，可生成丰富的训练数据集，用于训练查询优化器，使其根据查询特征预测最高效的执行计划。

步骤 3：生成准确标签 是为每个生成的执行计划分配估算成本，具体考虑以下因素：

- 索引扫描与表扫描：根据是否使用索引或执行全表扫描来确定代价差异。
- 哈希连接与嵌套循环连接：根据查询结构和数据分布估算不同连接方法的代价。
- 其他因素：CPU 成本、内存使用和磁盘 I/O。

准确的成本估算依赖于精确的基数，使用 GPT 来实现这一点将在后文进一步讨论。

2.3 挑战与讨论

2.3.1 生成查询的多样性。 实现 SQL 查询多样性的能力至关重要，因为这是开发有效生成器的关键。根据给定的数据，GPT 利用各种数据库特征和使用模式来优化查询生成和工作负载适配。

模式感知生成 使得给定一个模式后，GPT 能够构建一组遵循其结构的多样化查询，确保语法和语义正确性。

上下文感知生成 支持提供上下文感知的需求，从而创建针对特定用例工作负载量身定制的真实查询工作负载。指定后，GPT 会生成来自不同工作负载类别的混合查询（例如，复杂的聚合查询与简单的查找查询），以模拟基数估计器在实际中可能遇到的真实查询分布。当模型难以处理某些查询类型，需要更多这类查询示例来提升性能时，上下文感知查询生成也能发挥作用。

工作负载扩展 意味着在提供一个示例工作负载后，模型会从与给定领域相关的现有查询模式中学习，并对其进行优化和扩展。只需提供少量查询日志示例，GPT 就能生成与实际使用模式高度相似的查询，从而提升工作负载代表性，并带来查询性能洞察。

2.3.2 生成复杂查询模式。涉及多个连接或嵌套子查询的复杂查询，对生成模型构成了另一项挑战。要微调 GPT 并引导其朝正确方向生成，需要提供详细的提示词。例如，明确展示多种查询类型，包括包含嵌套连接和子查询的边缘情况，GPT 也能据此进行调整，生成此类真实且多样的查询模式。

2.3.3 查询计划多样性。查询优化面临的一大挑战是，单个查询可能对应多种执行计划。连接、扫描和聚合可以采用不同策略，而生成一套全面的策略十分困难。为解决这一问题，GPT 能够为同一查询生成多个备选计划。此外，还需要根据真实场景中的查询执行对这些计划进行验证，以确保其多样性和真实性。通过使用真实执行日志中的大量计划对模型进行微调，GPT 可以生成丰富多样的执行计划。

2.3.4 选择率与成本估算。默认情况下，GPT 无法准确执行计数或基数估算，因为它仅依据从大量文本数据中学到的模式进行运作，而非实际的数据分布或统计信息。由于准确的成本估算取决于基数估计值，因此查询优化中的标注也面临同样的问题。不过，模型可以利用选择率信息方面的指导，生成能够反映真实数据库访问模式、并与预期数据分布相符的查询。例如，提供表大小、列分布信息（如不同值计数、值频率）和谓词选择率等信息，在一定程度上有助于模型生成兼顾低选择率和高选择率模式的真实查询。

2.3.5 适配不同数据库。不同数据库系统的语法、优化器和执行策略各不相同。针对某个系统优化的查询，在另一个系统上可能并非最优。只要经过适当指示，GPT 就能适配多个数据库引擎。训练数据生成需要进行调整，以便生成适用于不同数据库引擎的工作负载，确保学习型模型能够在各种系统中泛化。这种适应性使生成的数据能够在多个平台上发挥更大作用，从而提升查询优化器的泛化能力。

2.3.6 扩展至大型数据集。生成大规模训练数据的一个关键问题是生成查询所需的计算成本。优秀的生成式模型应具备可扩展性，能够高效生成大量高质量数据。查询生成过程需要生成指定数量的 n 查询。由于模型的输出限制，当 n 足够大、单次响应无法容纳所有查询时，我们必须分批生成，以确保达到所需数量。在考虑查询优化时，随着潜在查询计划数量的增加，生成大量多样化训练数据的计算成本可能会变得很高。通过对现有查询稍作改动（例如修改连接条件），GPT 可以根据少量原始查询生成更多数据。这有助于扩展训练数据集，而无需承担过高的计算成本。尽管可以高效并行生成多样化查询，但要实现可扩展的标注仍然颇具挑战，需要采用传统索引方法或基数估算器，以快速准确地执行查询。

2.3.7 生成数据的保真度。虽然 GPT 能够生成语法有效的 SQL 查询，但确保这些查询能够代表真实的用例和查询模式，对于有效训练至关重要。针对这一挑战，可以使用真实世界查询日志对 GPT 进行领域特定微调，确保生成的查询反映典型的用户行为和数据库工作负载。若能从相似的连接、筛选条件和子查询、表达相同意图的不同形式，以及更深层次的语法和语义相似性等方面，检测**生成数据与真实世界数据之间的相似性**，将大有裨益。**领域专家评审或用户研究**也有助于评估生成数据是否反映了典型的数据库行为，例如人工检查部分查询、基数或执行计划。最后，使用生成数据训练模型，并用真实世界数据评估其**性能**，可以直接反映生成数据的质量。

3 可行性研究

为了初步了解生成式人工智能在为学习型数据库组件合成训练数据方面的效用，我们概述了关键指标，并介绍 GPT 处理各项任务的初步表现；实验使用了 OpenAI 的 GPT-4o，并采用经过修改、较为简单的 IMDB 模式和数据 [14]。

3.1 生成查询的多样性

3.1.1 模式感知生成。 为了分析 GPT 如何生成查询，我们首先提供 IMDB 模式，并使用一个简单的提示词为其生成一组多样化的查询。模型生成的各种查询**主要由等值谓词构成，其次是范围谓词、混合谓词，以及使用 IN 运算符的嵌套查询**。虽然优先生成没有连接或连接较少的简单查询，但也会包含更复杂的多表连接。不同查询中使用的表和列各不相同，模式给出的约束会用于连接谓词。在生成过程中，我们还可以限定每种谓词生成相同数量的查询，这对分类或测试任务尤其有用。

3.1.2 上下文感知生成。 使用 GPT 生成查询也具备上下文感知能力，例如，不同方法可能会侧重于不同的谓词条件，以展示其性能。定制提示词的示例如下：

> 生成一组带有不等谓词的 SQL 查询工作负载，用于测试 IMDB 数据集的基数估计器。该工作负载应包含带有简单和复合谓词不等式的查询。

从而生成专门针对这一用例的查询：

```
SELECT * FROM movies WHERE rating > 7.5; SELECT *
FROM movies WHERE release_year < 2000; SELECT *
FROM movies WHERE 时长 BETWEEN 90 AND 150 AND
评分 >= 6;
```

进一步来说，可以创建模拟特定用例的测试查询，例如：

> 生成一组 SQL 查询工作负载，用于测试 IMDB 数据集的基数估计器，方法是假设这是电影会计师的工作负载。

从而生成涉及会计师相关信息的查询，例如**收入、电影预算、时长和评分**。

3.1.3 工作负载扩展。给定样本工作负载且数据不足时，我们旨在生成额外查询作为补充。例如，若将前文所示的不等式查询作为样本提供，则会生成以下查询：

```
SELECT * FROM 电影 WHERE 时长 > 150AND 评
分 > 7.5; SELECT * FROM 电影 WHERE release_
year BETWEEN 1980 AND 2000 AND 评分 >= 6.
5;
```

模型通过改变数值范围并组合条件来扩展查询，同时保持原始意图。

3.2 选择性与代价估算

直观地说，当仅提供模式时，模型对列分布一无所知，只能依赖

提供给模型	仅等值谓词		仅不等谓词	
	非选择性	选择性	非选择性	选择性
边界与模式	0.02920	0.00330	0.70965	0.0382
样本与模式	0.0342	0.000679	0.6073	0.0133
直方图与模式	0.0328	0.0102	0.7143	0.0088

表1：不同查询类型的平均选择性

各种假设。例如，在针对汽车数据集要求执行高选择性查询时，GPT 会臆造出从全局角度看具有选择性的值，并将其纳入查询中，例如“劳斯莱斯”。因此，向模型提供额外信息至关重要。

为测试基于选择性生成内容的表现，我们提取标题表中的起始年份。首先，我们创建两个宽泛的提示词，其中包含数组样本：一个针对高选择性查询，另一个针对选择性不那么高的查询，形式如下：

生成 20 个具有高选择性的查询谓词，使查询结果数量较少。

模型根据其理解生成查询：等值谓词与高选择性相关，而宽范围谓词则与低选择性相关。由于 GPT 无法准确统计样本列表中单个元素的数量，因此选择性完全基于假设。

为进一步验证这一问题，我们要求它在不同设置下生成选择性和非选择性谓词（表1）。**边界与模式：** GPT 根据数据的最小值和最大值生成查询，但不了解指定范围内数值的分布。因此，它可能会在整个范围内均匀分布查询，也可能依靠启发式方法猜测常见值，例如假设最近年份出现得更频繁。对于等值查询，这通常会导致选择性较差，因为 GPT 不知道哪些值更常见或更罕见。对于不等式查询，GPT 会根据要求的选择性设定谓词边界。**样本与模式：** 如果样本具有代表性，GPT 就能推断出一些模式，例如频繁出现年份的簇。不过，它不会精确统计出现次数，而是根据在样本中观察到的模式进行估算。因此，它判断选择性的准确度可能有限。

直方图与模式： 与之前的情况类似，模型无法准确追踪数据，但能够识别哪些属于选择性查询，哪些属于非选择性查询。GPT 可能无法计算精确计数，但由于能够获取频率分布，因此在提供直方图时可以做出更好的估算。直方图有助于模型优先考虑常见值，但估算仍然基于模式，而非精确计数。

虽然 GPT 能区分高度选择性查询和非选择性查询，但无法生成预设范围内的值。当被要求生成特定选择性范围内的查询时，所有方法都无法给出合理结果，这就引出了一个问题：我们该如何标注数据?

3.3 扩展至大型数据集

最后但同样重要的是，生成合成工作负载的运行时间成本也必须在可接受范围内。为了获得大致数据，我们让 GPT 生成若干查询，并测量所需时间。

# 查询	10	20	30	40	50	100
每次查询的平均时间（毫秒）	486	377	380	415	309	310

表2：不同查询数量下的执行时间

如表2所示，生成更多查询会略微降低整体平均时间。执行时间还取决于查询大小、输入模式以及查询中包含的表数量等因素。

4 相关工作

数据库系统基准测试是科学出版物和工业性能研究的核心组成部分，常见的基准测试包括 TPC-H [1], SSB [18], 连接顺序基准测试（JOB） [14], 或 TPC-H Skew [3]。

Hilprecht 和 Binnig [10] 提出了零样本学习，以训练数据库的学习型组件，避免学习型方法以往所需的高成本（重新）训练 [19]。尽管他们的目标与我们相同，但采用的角度截然不同。他们提出基于数据和工作负载进行学习，并为此设计了一个新的基准测试，其中通过不同的查询模式实现了一个工作负载生成器。与此不同，我们建议将工作负载的生成交给大语言模型——我们不提出任何新的学习方法。另一个相关领域是自动化数据库调优，近期的研究利用机器学习方法调整数据库的各个方面，例如索引创建、缓冲区大小等 [23], 、使用优化器提示 [21], , 或利用（大型）语言模型理解参数的含义并遵循其建议 [2, 8]。这些工作都没有专门针对数据库系统内部学习型组件的训练。

要利用大语言模型生成工作负载，其理解数据库模式并生成 SQL 查询的能力至关重要。文本到 SQL [5, 11]的研究正转向使用大语言模型 [6, 9, 15]，而 GPT 等近期推出的现成大语言模型已经展现出令人瞩目的、近似人类的模式理解能力和查询意图理解能力，在文本到 SQL 任务 [7]中也表现出色。

5 结论与展望

我们提出利用生成式人工智能自动生成数据库工作负载，以支持学习型数据库组件的开发与优化，从而增强其适用性。通过一项初步可行性研究，我们指出了关键需求与挑战，并证明在提供精心设计的提示词后，GPT 等模型能够有效协助处理数据库模式和 SQL，生成查询。

虽然 GPT 模型能够在各种约束条件下有效生成多样化的 SQL 查询，但生成符合特定选择率要求的查询仍是一项重大挑战。要实现对选择率的精细控制，仅靠语言生成还不够——还需集成索引结构，以引导生成过程，确保选择率过滤和标签分配正确。另一个前景广阔的方向是生成参数化查询，根据额外的统计信息系统地引入选择率变化，从而更好地控制生成工作负载的多样性和代表性。

参考文献

[1] [n.d.]。《TPC-H 基准测试》。https://www.tpc.org/tpch/。 [在线；访问于 03-April-2025]。[2] Sibei Chen、Ju Fan、Bin Wu、Nan Tang、Chao Deng、Pengyi Wang、Ye Li、Jian Tan、Feifei Li、Jingren Zhou 和 Xiaoyong Du。2025年。使用检索增强语言模型自动调试数据库配置。*ACM 管理数据会议录*3, 1 (2025), 13:1–13:27。[3] Alain Crolotte 和 Ahmad Ghazal。2011。将偏斜引入 TPC-H 基准测试。载于*TPCTC*（计算机科学讲义，第 *7144* 卷）。Springer，137–145。[4] Angjela Davitkova、Evica Milchevski 和 Sebastian Michel。2020。ML-Index：一种用于点查询、范围查询和最近邻查询的多维学习型索引。载于*EDBT*。OpenProceedings.org，407–410。[5] Naihao Deng、Yulong Chen 和 Yue Zhang。2022。文本到 SQL 的最新进展：现状与展望综述。载于*COLING*。国际计算语言学委员会，2166–2187。[6] Ju Fan、Zihui Gu、Songyue Zhang、Yuxin Zhang、Zui Chen、Lei Cao、Guoliang Li、Samuel Madden、Xiaoyong Du 和 Nan Tang。2024。结合小型语言模型和大型语言模型实现零样本 NL2SQL。*VLDB* 会议录 17, 11 (2024), 2750–2763。[7] Dawei Gao 、 Haibin Wang、Yaliang Li、Xiuyu Sun、Yichen Qian、Bolin Ding 和 Jingren Zhou。2024。由大型语言模型赋能的文本到 SQL：基准测试评估。*VLDB* 会议录17, 5 (2024), 1132–1145。[8] Victor Giannakouris 和 Immanuel Trummer。2025。 *λ*-Tune：利用大型语言模型自动调优数据库系统。*ACM 管理数据会议录*3, 1 (2025), 2:1–2:26。[9] Zihui Gu、Ju Fan、Nan Tang、Lei Cao、Bowen Jia、Sam Madden 和 Xiaoyong Du。2023。使用结构与内容提示学习进行少样本文本到 SQL 翻译。*ACM 管理数据会议录*1, 2 (2023), 147:1–147:28。[10] Benjamin Hilprecht 和 Carsten Binnig。2022。用于开箱即用学习型成本预测的零样本成本模型。*VLDB* 会议录15, 11 (2022), 2361–2374。[11] George Katsogiannis-Meimarakis 和 Georgia Koutrika。2023。文本到 SQL 深度学习方法综述。*VLDB* 杂志32, 4 (2023), 905–936。[12] Andreas Kipf、Thomas Kipf、Bernhard Radke、Viktor Leis、Peter A. Boncz 和 Alfons Kemper。2019。学习型基数：使用深度学习。载于*CIDR*。www.cidrdb.org。[13] Tim Kraska、Alex Beutel、Ed H. Chi、Jeffrey Dean 和 Neoklis Polyzotis。2018年。学习型索引结构的意义。载于*SIGMOD*会议。ACM，489–504。[14] Viktor Leis、Andrey Gubichev、Atanas Mirchev、Peter A. Boncz、Alfons Kemper 和 Thomas Neumann。2015年。查询优化器究竟有多好？*VLDB* 会议录9，3（2015），204–215。[15] Xinyu Liu、Shuyu Shen、Boyan Li、Peixian Ma、Runzhi Jiang、Yuyu Luo、Yuxin Zhang、Ju Fan、Guoliang Li 和 Nan Tang。2024年。大型语言模型驱动的NL2SQL综述：我们身处何处，又将走向何方？*CoRR* abs/2408.05109（2024）。[16] Ryan Marcus、Parimarjan Negi、Hongzi Mao、Chi Zhang、Mohammad Alizadeh、Tim Kraska、Olga Papaemmanouil 和 Nesime Tatbul。2019年。Neo：一种学习型查询优化器。*VLDB* 会议录12，11（2019），1705–1718。[17] Parimarjan Negi、Ziniu Wu、Andreas Kipf、Nesime Tatbul、Ryan Marcus、Sam Madden、Tim Kraska 和 Mohammad Alizadeh。2023年。变化的工作负载下鲁棒的查询驱动基数估计。*VLDB* 会议录 16，6（2023），1520–1533。[18] Patrick E. O'Neil、Elizabeth J. O'Neil、Xuedong Chen 和 Stephen Revilak。2009年。星型模式基准测试与增强事实表索引。载于*TPCTC*（计算机科学讲义，第*5895*卷）。Springer，237–252。[19] Ji Sun 和 Guoliang Li。2019年。端到端学习型成本估计器。*VLDB* 会议录 13，3（2019），307–319。[20] Immanuel Trummer、Junxiong Wang、Ziyun Wei、Deepak Maram、Samuel Mose- ley、Saehan Jo、Joseph Antonakakis 和 Ankush Rayabhari。2021年。SkinnerDB：通过强化学习进行遗憾有界查询评估。*ACM数据库系统汇刊* 46，3（2021），9:1–9:45。[21] Lucas Woltmann、Jerome Thiessat、Claudio Hartmann、Dirk Habich 和 Wolf- gang Lehner。2023年。FASTgres：使学习型查询优化器提示有效。*VLDB* 会议录 16，11（2023），3310–3322。[22] Zongheng Yang、Eric Liang、Amog Kamsetty、Chenggang Wu、Yan Duan、Xi Chen、Pieter Abbeel、Joseph M. Hellerstein、Sanjay Krishnan 和 Ion Stoica。2019年。深度无监督基数估计。*VLDB* 会议录 13，3（2019），279–292。[23] William Zhang、Wan Shen Lim、Matthew Butrovich 和 Andrew Pavlo。2024年。Holon方法：借助机器学习和合成原型动作，在自驱动数据库管理系统中同时调优多个组件。*VLDB* 会议录 17，11（2024），3373–3387。