计算机研究与发展
Journal of Computer Research and Development
DOI: 10.7544/issn1000-1239.202220649
61(2): 413−427, 2024

基于机器学习的基数估计技术综述

岳文静[1,2] 屈稳稳[1] 林 宽[3] 王晓玲[1]
[1](华东师范大学计算机科学与技术学院 上海 200062)
[2](华东师范大学上海智能教育研究院 上海 200062)
[3](中国科学院空天信息创新研究院 北京 100190)
(wjyue@stu.ecnu.edu.cn)

Survey of Cardinality Estimation Techniques Based on Machine Learning

Yue Wenjing[1,2], Qu Wenwen[1], Lin Kuan[3], and Wang Xiaoling[1]
[1](*School of Computer Science and Technology*, *East China Normal University*, *Shanghai* 200062)
[2](*Shanghai Institute of AI for Education*, *East China Normal University*, *Shanghai* 200062)
[3](*Aerospace Information Research Institute*, *Chinese Academy of Sciences*, *Beijing* 100190)

Abstract Cardinality estimation is the basis and core of query optimizer for the database management system (DBMS). With the development of artificial intelligence (AI) technology, AI technology has shown superior performance in data processing and extracting the relationship from the data. In recent years, the research of the cardinality estimation method based on machine learning has made significant progress and received wide attention from the academic community. Firstly, we introduce the technical background and development status of cardinality estimation methods based on machine learning. Secondly, we give the definition and the feature encoding technology of the related concepts of cardinality estimation. Then, we expound on the classification structure of cardinality estimation technology from two aspects: traditional cardinality estimation and cardinality estimation based on machine learning. Then, we further subdivide cardinality estimation based on machine learning into three types of cardinality estimation techniques: query-driven, data-driven, and hybrid models. Then, we focus on analyzing the modeling flow, typical methodologies, and characteristics of each type of model. In addition, we analyze and summarize the application of cardinality estimation in SQL and NoSQL. Finally, we discuss the challenges and future research directions on cardinality estimation methods based on machine learning.

Key words database; cardinality estimation; machine learning; query optimization; neural network

摘 要 基数估计是数据库关系系统查询优化器的基础和核心.随着人工智能技术的发展,其在数据处理、提取数据之间的关系等方面显现出优越的性能.近年来,基于机器学习的基数估计技术取得了显著的进展,受到了学术界的广泛关注.首先总结了基于机器学习的技术估计技术的发展现状,其次给出了基数估计的相关概念及其特征编码技术.接着建立了基数估计技术的分类体系.在此基础上,进一步将基于机器学习的基数估计技术细分为查询驱动、数据驱动和混合模型这3类基数估计技术.然后重点分析了每一类

收稿日期: 2022-07-24; **修回日期**: 2023-04-19
基金项目: 国家重点研发计划项目 (2021YFC3340700); 国家自然科学基金重点项目 (62136002); 上海市科委重点项目 (20DZ1100300); 国家自然科学基金项目 (61972155)
This work was supported by the National Key Research and Development Program of China (2021YFC3340700), the Key Program of the National Natural Science Foundation of China (62136002), the Science and Technology Commission of Shanghai Municipality (20DZ1100300), and the National Natural Science Foundation of China (61972155).
通信作者: 王晓玲(xlwang@cs.ecnu.edu.cn)

技术的建模流程、典型技术和模型特点,并对其在 SQL 和 NoSQL 中的应用进行了分析和总结.最后讨论了基于机器学习的基数估计技术面临的挑战和未来的研究方向。

关键词 数据库;基数估计;机器学习;查询优化;神经网络

中图法分类号 TP392

基数估计(cardinality estimation)是数据库执行查询优化中的一个重要环节,用于估计在数据库中执行查询后返回的结果数. 理论上,只要提供了准确的基数估计和物理计划执行代价,并可以在巨大搜索空间中有效地枚举计划,数据库就可以在合理的时间内制定出最佳查询计划. 传统的基数估计技术一般采用像直方图等数据结构的统计方法来表示数据表上的数据分布,或者通过采样的方法估计数据的分布. 但实际上,由于复杂谓词和对多个数据表估计等复杂查询的固有难题,以及数据日益增加的复杂性,数据库的查询场景越来越复杂,其对于基数估计技术在处理多列、多谓词查询以及多表连接等情况下的能力要求逐渐增强. 尽管传统的基数估计技术可以在短时间内给出数据集的统计结果,但由于其难以兼顾空间开销与估计准确率、难以处理复杂数据分布等特性,使其在面对复杂困难的查询场景时,基数估计的性能仍有待提升.

近年来,鉴于人工智能技术在复杂的数据处理和模型学习方面的能力,数据库研究人员在基数估计技术中引入了人工智能技术,提出了基于机器学习的基数估计技术. 如何建模查询与基数之间的映射关系,以及如何建模数据、列属性与数据表之间的映射关系是目前研究中关心的 2 个核心问题. 在已有的研究中,基于机器学习的基数估计方法一般分为 2 类:一类方法为查询驱动的基数估计方法[1-14],此类方法直接建模查询负载和基数之间的关系;另一类方法为数据驱动的基数估计方法[15-21],此类方法通过拟合数据库中的数据联合分布函数来计算基数.

最新的研究[22-25]中对比了传统的基数估计算法和基于机器学习的基数估计算法,展示了基于机器学习的基数估计技术可以很好地提升基数估计的准确性. 但它们也继承了人工智能算法建模的一些不可解释性、鲁棒性差等问题,距离实际落地应用仍有很大的差距.

基数估计技术是查询处理中的核心技术之一,也是学术界一直关注的研究热点. 文献 [25] 主要介绍了传统的基数估计技术和查询驱动的基数估计技术. 文献 [22] 详细地介绍了传统的基数估计相关技术,并从是否使用了监督学习技术的角度讨论了基于机器学习的基数估计的相关技术,同时对相关基数估计技术中编码的内容进行了简单说明. 文献 [24] 侧重于比较在无数据更新和数据频繁更新的查询场景下,传统的基数估计技术与基于机器学习的基数估计技术的性能表现. 同时从技术实际落地角度出发,给出了基于学习的基数估计技术未来的研究方向. 文献 [23] 为基于学习的基数估计技术探索了统一的设计空间,主要介绍了查询驱动与数据驱动的基数估计技术,同时在准确率和效率 2 个方面对相关技术进行了分析比较. 本文在已有综述研究的基础上:1)增加了针对基于机器学习的基数估计特征编码技术的对比、分析和总结;2)基于文献 [23] 中提供的设计空间,以及对基于机器学习的基数估计相关技术的总结,本文提炼了建模查询驱动和数据驱动 2 类不同映射关系的核心工作流程,进一步在传统的 SQL 查询和 NoSQL 应用中分析了查询驱动、数据驱动的基数估计技术,同时补充了最新的前沿技术. 由于混合模型的基数估计技术的兴起,本文增加了对这类基数估计技术的介绍与分析,从而进一步完善基数估计技术的分类体系.

本文的组织结构包含 4 个方面:1)全面阐述数据库中的基数、基数估计以及基于机器学习的基数估计的定义,并分析了其使用的特征编码技术及基数估计技术中的分类体系. 2)根据提炼出的查询驱动的基数估计和数据驱动的基数估计的建模流程,分别介绍了这 2 类具有代表性的基于机器学习的基数估计技术,并补充了混合模型的基数估计建模流程以及相关技术. 3)在查询驱动的基数估计技术中整理了其在 NoSQL 的应用技术. 4)讨论了基于机器学习的基数估计技术面临的挑战以及未来的研究方向.

1 基数估计定义及分类

1.1 基数估计

给定 SQL 查询语句和数据库 D, 基数(cardinality)是在数据库 D 中执行查询 q 返回的结果行数,记为 $C(q|D)=|D'|$, 其中 D' 表示查询结果,而基数估计则表

示在不执行查询 q 的情况下，对基数进行估计[23]. Leis 等人[26]证明了代价估计的性能主要与基数估计的精度有关，而代价估计是数据库查询优化中的关键步骤. 因此基数估计对查询优化器产生最优的查询计划至关重要.

1.2 基于机器学习的基数估计

给定 SQL 查询 q、数据库 D，基于机器学习的基数估计是要学习 q 和基数 C 之间的一个映射函数[23]，或者学习一个拟合数据库中数据分布的函数[23]，输入 q 后函数返回估计的基数 $\hat{C}$，不同的任务使得函数的表现形式不同. 基于机器学习的基数估计技术的目标就是尽可能地使 $\hat{C}=C$.

1.3 特征编码技术

将 SQL 语句或者使用的数据表编码为特征向量作为基数估计模型的输入是基于机器学习的基数估计技术的基础. 本文对基于机器学习的基数估计技术中的特征编码技术进行了 2 个维度上的分类，一类为特征编码的类型，分为谓词(运算符、查询值与列序号(id)、连接类型、数据库元数据、数据表数据和其他属性的特征. 另一类为特征的编码方式，共分为 3 类. 其中独热(One-Hot)编码、嵌入(Embedding)编码是最常用的编码方式，其余基于特定查询属性的编码方式归结为其他类型的编码方式. 表 1 将已有研究分别对应到特征编码技术的 2 个分类中. 下面本文将根据特征编码方式对已有研究中编码的特征类型进一步归纳总结.

1.3.1 One-Hot 编码

One-Hot 编码将查询属性特征转换为同一维度的向量表示，并使用 0 或 1 表示. 基于神经网络(neural network, NN)的模型 NN_{ST}[3] 使用 One-Hot 编码了查询计划，其中将查询计划根据数据库属性信息和单一操作运算符支持的谓词(即支持等值谓词编码，对于 1 维范围选择谓词，仅编码了查询值)编码为子查询的特征向量，其中数据库属性信息包括最大值、最小值、基数和 1 维等宽直方图表示. 不同的研究对于查询计划的属性特征定义不同，其中模型 MSCN[5] 编码了查询语句中表的集合、连接的集合和谓词集合的表示，并作为模型的输入. 值得注意的是，谓词集合中仅使用了 One-Hot 编码列 id 和运算符 id，查询值使用的是归一化的数值. 而文献 [6] 编码了查询语句中的表 id、选择谓词以及等值连接谓词，同样使用了归一化的方式将范围选择谓词中的查询值进行编码. 该文献中提出的模型支持数据表的等值连接，通过选用不同的编码特征和不同的机器学习模型，使其可以适用于文献 [6] 中定义的查询类型. 文献 [8] 由于同时可以估计查询成本，因此使用了 One-Hot 编码物理操作，包括连接类型和 3 类算子，同时也对数值型的查询谓词进行了编码.

1.3.2 Embedding 编码

在属性个数较多的情况下使用 One-Hot 编码，特征空间维度巨大且编码稀疏，进而计算代价较大. 为了解决 One-Hot 编码的缺陷，Embedding 编码将稀疏的查询特征表示变为低维稠密空间的特征表示. 文献 [17,19] 均编码了数据表中的数据作为模型的输入向量，对于不同取值数较大的列，均使用 Embedding 编码来将表内数据编码为一个稠密向量，否则使用 One-Hot 编码. 模型 GL(global-local)[27] 将查询、提出的阈值编码为向量，同样使用了 One-Hot, Embedding 编码等方式将高维图片或者文本等数据编码为数据向量，然后使用神经网络将向量进一步转换为模型输入需要维度的特征向量.

1.3.3 其他编码方式

除了 1.3.1 节和 1.3.2 节中所述的 2 种常见的编码方式之外，一些研究对特定的查询特征属性提出了特定的编码方式. 模型 Local NN[4] 使用了 2 种方式进行编码，该模型使用 One-Hot 编码技术编码运算符，对于查询中的值，若其为数值型则使用归一化公式进行编码，否则，使用字典编码将非数值型的值转换为数值型. 文献 [7] 利用语义模板定义了查询的多种特征，并直接使用特征的变量值作为查询的多维特征向量. 其中，多种特征包括查询的属性、运算符、查询数值、模型参数、聚合和用户自定义函数(user-defined function, UDF)的参数等. 文献 [8] 提出使用字符串编码技术编码字符型的查询谓词，使用位图来编码样本位图，即使用固定大小的向量来判断每位对应的元组是否满足查询节点的谓词. 文献 [9] 中提出的 2 类模型，分别为树模型和 NN 模型. 将范围查询中 d 个数字属性的范围上下界直接拼接为 1 个 $2d$ 的元组作为范围特征. 此外，还将单属性直方图中的信息生成查询的选择度估计作为回归模型的额外特征. 模型 DLM[18] 将数据表中的一行数据看作一个序列，根据值 id 将每个值编码为一个二进制向量. 而文献 [28] 自动确定规模的极端梯度提升(extreme gradient boosting, XGBoost)模型使用数值转换的方式将查询范围谓词和分类谓词转换为十进制数字作为查询特征，并将计算查询中单表的选择度估计值作为选择度特征.

1.4 基数估计技术分类体系

基数估计技术根据是否使用了机器学习算法分

Table 1 Feature Encoding Techniques Based on Machine Learned Cardinality Estimation Techniques

表 1 基于机器学习的基数估计技术的特征编码技术

编码方式	模型	特征类型				
		谓词	连接类型	数据库元数据	数据表数据	其他
One-Hot 编码	NN_{ST} [3]	等值连接谓词和范围选择谓词中的查询值		最大值、最小值、基数和 1 维等宽直方图表示		
	Local NN [4]	每个谓词的单一运算符<，=，>				
	MSCN [5]	列 id、运算符 id	连接类型 id	所有数据表的 id		
	文献 [6]	等值连接谓词和单向范围选择谓词		所有数据表的 id		
	文献 [8]	数值型的查询谓词中的列 id 和运算符 id	连接类型 id			扫描、聚合和排序算子
	Naru [17]				不同取值数较小的列	
	NeuroCard [19]				不同取值数较小的列	
	GL（global-local）[27]	相似性查询中高维图片或者文本等查询值				相似距离阈值
Embedding 编码	Naru [17]				不同取值数较大的列	
	NeuroCard [19]				不同取值数较大的列	
	GL [27]	相似性查询中高维图片或者文本等查询值				相似距离阈值
其他编码	Local NN [4]	使用归一化公式或字典编码查询值				
	文献 [7]	运算符、查询值使用不超过参数域范围的值编码为数值				模型参数、聚合、UDF 的参数
	文献 [8]	字符串编码技术编码字符串型的查询谓词				使用位图编码样本位图
	Tree-based ensembles/NN [9]	范围查询的上下界				选择度估计
	DLM [18]				将每列数据的序号编码为二进制	
	自动确定规模的 XGBoost [28]	范围查询的上下界与十进制的分类谓词查询值				选择度估计

为传统的基数估计技术和基于机器学习的基数估计技术 2 类，其分类体系如图 1 所示.

传统的基数估计技术一般被分为基于统计的和基于采样的方法. 基于统计的方法[29-35]的核心是使用某种数据结构，例如直方图[29-31]、数据画像[32-35]来拟合表上的数据分布，其中，文献 [29] 采用了直方图来近似数据库中的数据分布，这类基于直方图的统计方法估计精度较高，但是由于其内存占用大，所以检索速度较慢. 该类方法只考虑了拟合单列的数据分布，没有考虑多列属性之间的相关性；文献 [32] 采用数据画像来估计数据集中不同元素的个数，使用随机哈希函数并基于随机假设，可以将任意数据集映射成服从均匀分布的随机值序列从而进行估计. HyperLogLog[33] 基于基数估计器 LogLog，使用调和平均数降低了方差. 相比于基于直方图的估计方法，基于数据画像的方法通常探索一个性能良好的哈希函数来拟合数据分布，占用内存较小、检索速度更快. 但是这类方法仍旧仅针对单列的数据估计其基数，不适用于现实中复杂的查询应用. 基于采样的方法[36-40]的核心是通过从原始数据集中采样到小数据集来估计数据的整体分布，以反映不同表之间的关联关系. 不同于基于统计的基数估计方法，基于采样的基数估计方法不依赖于特定的假设，通过对不同的表按照一定的采样方式采集元组，并进一步利用这些元组估计基数. 但如何高效地在一个规模较大的数据集上进行采样，当底层数据发生更新后如何进行重采样，如何支持非等值连接仍然是需要解决的问题[22].

由于传统的基数估计技术计算需要存储直方图、位图、采样的元组等信息，会占用较大的存储空间；而基于机器学习的基数估计技术仅需要存储学习映射函数的模型，因此相较于传统基数估计技术，其模型占用空间小. 基于统计的基数估计技术适用于拟合单列的数据分布，而在处理任意多列组合数据之间的复杂关系，其能力较弱. 基于采样的方法可以有效支持小规模数据表的查询. 但随着数据的日益增加，基于采样的基数估计技术在大规模数据表的复

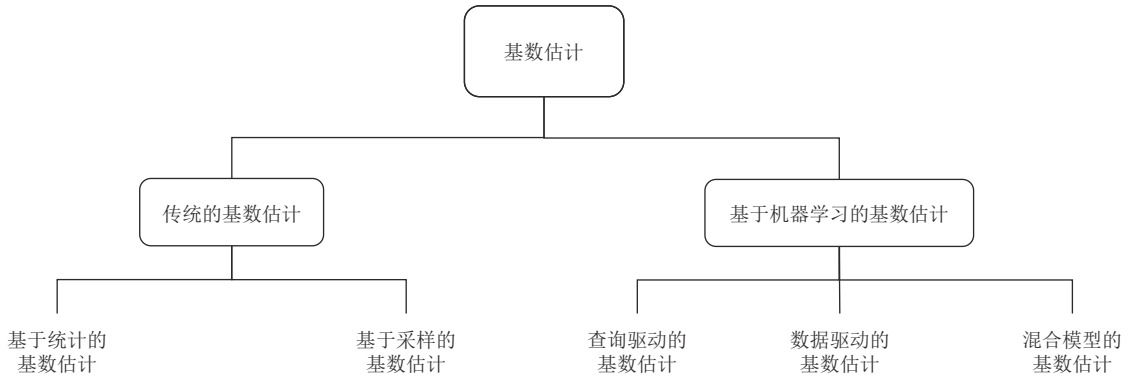


Fig. 1 The classification system of cardinality estimation

图 1 基数估计分类体系

杂查询场景中需要消耗大量空间存储采样的元组，同时有效采样元组会随着多表复杂的连接而减少，损失估计的性能. 因此，传统的基数估计技术在处理涉及多列、多谓词的查询，以及多表连接等复杂困难的查询场景时，常会导致较高的预测偏差[23]. 由于机器学习技术可以学习到非线性数据之间的复杂关系，因此基于机器学习的基数估计技术可以更好地表达查询特征以及拟合数据的复杂分布，进而能够支持较为复杂的查询场景.

基于机器学习的基数估计技术由于基数估计精度高、占用空间小、拟合复杂数据关系能力强等特点，受到了研究者们的广泛关注与青睐. 根据已有工作，本文将基于机器学习的基数估计技术在主流数据库中的应用分为 3 类，分别为查询驱动的基数估计、数据驱动的基数估计和两者混合模型的基数估计. 此外，查询驱动的基数估计在 NoSQL 中也得到了广泛应用.

2 查询驱动的基数估计技术

查询驱动的基数估计技术的核心思路是学习查询 q 和基数 C 之间映射的一个回归函数 $f_\theta(q,C)$，其中 θ 表示模型需要学习的参数. 查询驱动的基数估计技术通常使用不同的编码方式、编码查询负载的不同特征属性，基于不同的监督学习算法模型建模查询、列属性和表三者之间的关系，以支持不同数据量的查询操作. 首先，本文基于已有的查询驱动的基数估计技术，归纳出查询驱动的基数估计的一般性建模流程；然后对不同查询驱动的基数估计相关模型进行了详细的介绍，其中包括了在 NoSQL 中的相关研究；最后对相关模型进行了对比和总结.

2.1 查询驱动的基数估计建模流程

查询驱动的基数估计技术的核心思想是建立一个查询驱动的基数估计模型，其一般性的建模流程如图 2 所示，分为输入、输出和模型建立 3 部分[23-24].

1）输入

输入数据集 $DS_{\text{train/test}}\{\langle q,C\rangle\}$、数据库 D 以及模型约束参数 ϕ. 例如，对于查询驱动的代表模型 MSCN[5]，其输入为数据集 $DS_{\text{train/test}}$ 和数据库 D. 其中，给定 SQL 查询 q 经过查询解析器后构建出的数据集作为训练数据，同时将 q 对应的真实基数 C 作为标签.

2）输出

在数据库 D 中执行查询 q 后估计的基数 $\hat{C}$. 在得到估计的基数后，经过基于代价模型的代价估计，使得数据库可以制定查询计划，进而完成查询优化.

3）模型建立

①模型训练. 如图 2 中粗实线条箭头所示，模型共包含 2 个模块：查询特征编码器和监督学习模型. 首先模型使用查询特征编码器，根据不同的任务或者模型需求，将输入的查询 q 解析为具有固定维度 φ 的查询特征向量 $\boldsymbol{Q}$、谓词特征等，其中 $|\boldsymbol{Q}|=\varphi$，查询特征编码器中的编码技术如表 1 所示. 将 $\boldsymbol{Q}$ 与真实基数 C 构建为输入对 $\langle\boldsymbol{Q},C\rangle$ 输入到监督学习模型中进行训练，在训练过程中构造特定的损失函数，不断优化模型的学习参数 θ.

本文以模型 MSCN 为例来解释查询驱动的基数估计模型训练的流程. 首先将连接查询 q 分为 3 部分，分别表示查询语句中表的集合、连接的集合和谓词的集合，从而将查询、列属性和表关联起来. 然后将这 3 个集合分别输入到 3 个 2 层的神经网络中得到 3 个对应的特征向量. 3 个特征向量拼接后得到的查询特征向量与真实的基数一起输入到 2 层的多层感知机中，并使用梯度下降算法进行训练.

②模型推理. 如图 2 中细实线条箭头所示，基数估计在推理阶段同样需要经过①中的 2 个模块. 不同

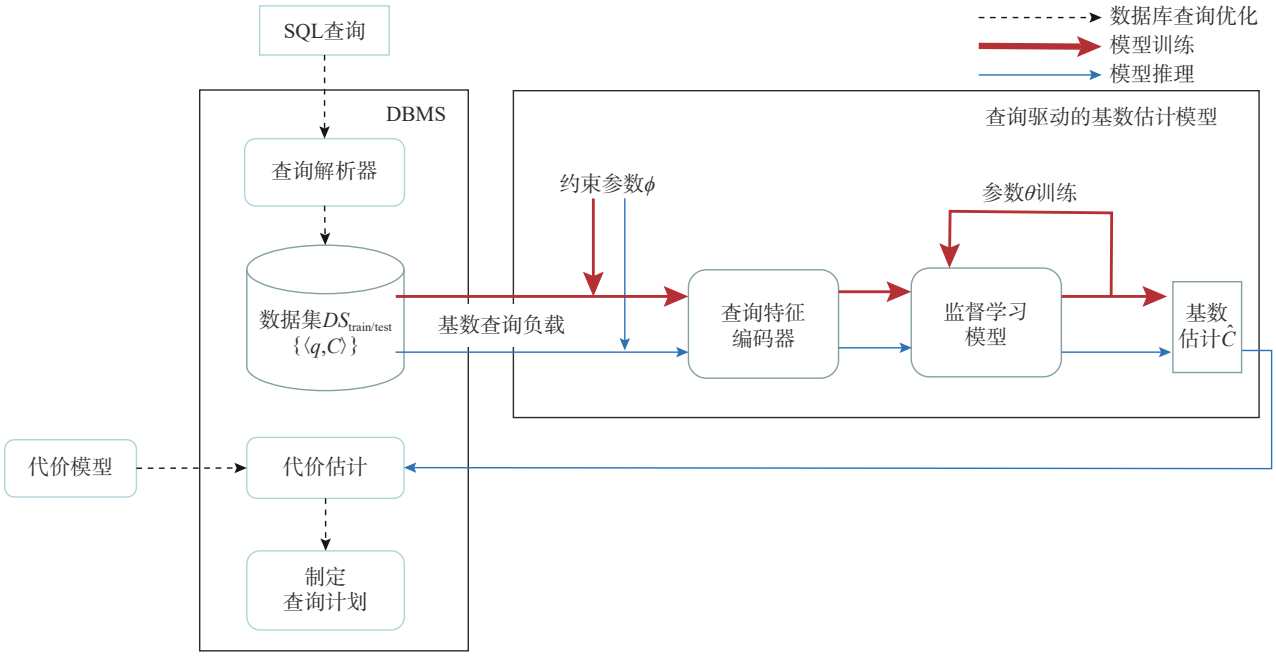


Fig. 2 General modeling flow for query-driven cardinality estimation

图 2 查询驱动的基数估计的一般性建模流程

的是, 在模型推理阶段, 模型参数 θ 已知, 数据库可以直接通过已经训练好的模型估计查询 q 的基数.

同理以模型 MSCN 为例来解释查询驱动的基数估计模型推理的流程. 当输入一个新的查询 q 后, MSCN 同样将 q 分为 3 部分并利用训练好的 3 个神经网络分别编码这 3 个部分为特征向量, 拼接这 3 个特征向量后输入到训练好的多层感知机中得到最终估计的基数 $\hat{C}$.

2.2 查询驱动的基数估计相关模型

在目前的研究[1-9, 28]中, 许多研究人员根据不同的查询需求和基于不同的监督学习方法建立了不同的基数估计模型. 本文根据监督学习类型将已有研究分为 2 类, 其中文献 [1−6] 是基于神经网络算法的基数估计模型, 文献 [7−9, 28] 是基于树结构的基数估计模型. 其次, 本文分析了查询驱动的基数估计在 NoSQL[27, 41-47] 上的应用研究.

2.2.1 基于神经网络算法的基数估计模型

神经网络优秀的大规模数据拟合能力和非线性建模能力可以带来更好的估计精度. 因此, 许多研究将神经网络引入到查询回归模型中, 用于处理不同类型的查询负载. 在基于神经网络算法的基数估计技术中, Lakshmi 等人[1]将神经网络应用到用户自定义函数的基数估计中, 并将反向传播算法应用到基数估计技术. 此研究是将基数估计应用到神经网络的初步探索. 随着深度学习的发展 Ortiz 等人[3]提出使用深度强化学习对子查询的属性进行编码, 将执行计划转换为强化学习中的动作表示, 并使用一个神经网络 NN_{ST} 得到在新动作下新的子查询计划的表示. 同时使用另一个神经网络预测每个状态下子查询的基数, 并根据预测的基数利用强化学习优化查询. 2019 年, Ortiz 等人[6]提出使用循环神经网络(recurrent neural network, RNN)模型预测查询的基数, 该模型将一个 SQL 语句看作一个序列. Ortiz 等人[6]的这 2 个研究可以很好地适配不同长度、不同类型的查询操作, 但这 2 个技术仅专注于解决左深度支持的查询计划, 而对于一些其他结构支持的查询计划无法直接适用.

为了解决使用神经网络方法建模整个数据库模式带来的数据稀疏问题, Woltmann 等人[4]使用神经网络为不同且固定的查询连接条件训练了一个局部模型 Local NN. 该局部模型的思想使基数估计模型可以更好地表示出数据之间的相关性, 然而这种训练方式消耗较大. 不同于该模型, Kipf 等人[5]基于卷积神经网络算法, 同时基于采样的思想来学习多表谓词关联性, 并为整个查询训练了一个全局模型 MSCN 用于估计连接查询的基数以减小训练消耗.

2.2.2 基于树结构的基数估计模型

树结构由于其模型参数较小、运行时间快的特点被广泛关注. 为了使模型具有学习能力, 同时模型参数或运行时间又不会太大, 树结构这种模型结构被应用到基数估计模型中用于增量学习查询负载的特征属性. 在基于树结构的基数估计中, Malik 等人[7]率先

提出了一种“黑盒”的方法来估计查询基数，在不了解查询执行基数和数据分布的情况下提供了准确的估计. 该黑盒的主要思想是将查询分组到固定的句法组中，并使用基于树状模型等机器学习技术来学习每个句法组的查询结果大小的分布，并编码了查询的属性、运算符、模型参数、聚合和用户自定义函数的参数，该方法显著提高了 Open SkyQuery 中的缓存性能.

为了适应不同结构的查询负载，Sun 等人[8] 提出了一个基于树结构模型的端到端成本估计学习框架，该框架可以同时估计成本和基数. 该研究提出了有效的特征提取和编码技术. 其中，在特征提取时考虑了查询和物理操作，并将特征嵌入到树结构模型中；在特征编码时提出了一种有效的字符串值编码技术，可以提高谓词匹配的泛化能力. 同时，该研究提出了一种基于模式（pattern）的方法，用于覆盖所有的字符串值并编码其嵌入.

Dutt 及其团队分别于 2019 年[9] 和 2020 年[28] 提出了基于 XGboost 算法的基数估计模型. 文献 [9] 为多列范围查询提出了一种轻量级的选择度估计方法，并提出了基于树模型和基于神经网络模型预测的选择度. 该研究首先将范围查询编码为范围特征，并使用启发式的基数估计器为单属性直方图中的信息计算查询的选择度来作为额外特征，并将 2 类特征输入到树结构或神经网络结构的回归模型中进行训练. 为解决训练集收集耗时长、模型构建时间长和为大型数据集计算选择度标签代价大的问题，从构建训练数据的角度上，文献 [28] 提出了一个基于霍夫丁（Hoeffding）不等式的生成近似选择度标签策略，用于增量标记数据. 该研究首先随机选择少量的训练数据并标记，然后将查询特征和单表的选择度特征输入到 XGBoost 模型中计算查询的选择度，若模型的 Q-error 值低于阈值的置信区间，则模型停止训练. 此研究在使用回归模型预测查询基数的同时自动确定了训练数据规模大小.

2.2.3 NoSQL 中查询驱动的基数估计模型

以上的研究聚焦于传统的 SQL 查询语句，模型适用于低维数据的处理，但对于 NoSQL 中高维的图片、文本数据以及复杂的 SPARQL 查询，以上模型不能直接适用. Sun 等人[27] 将 SQL 查询对象扩展到了图片、文本等更高维的数据，基于神经网络算法并结合全局和局部模型的思想，针对高维数据的相似性查询和连接任务设计了一个基数估计模型 GL，用于平衡模型数量和模型参数规模. 该模型提出了查询分割和数据分割 2 个策略，尽管仅使用了简单的神经网络模型进行特征编码和基数估计，但其效果显著. 该模型有效解决了神经网络方法中的数据饥饿问题，提高了高维数据在相似性查询和连接操作中的准确率. 虽然此模型相对于以往基于神经网络的模型，模型规模和参数量下降，但该模型的训练和预训练的时间仍然较长.

此外，传统 SQL 查询的基数估计模型同样不能直接适用于语义网底层数据模型资源描述框架的查询语言 SPARQL. 基于此挑战，Hasan[41] 提出了利用机器学习技术预测 SPARQL 查询的执行时间. 该工作采用支持向量回归模型作为基数估计模型，并使用图编辑距离（graph edit distance, GED）方法提取 SPARQL 查询辅助的特征属性. 但是，当训练数据集很大的前提下，GED 的计算非常耗时，且系统规模较大.

针对这一问题，Zhang 等人[42] 首先定义了 SPARQL 的代数特征和基本图模型 2 类特征，该特征分别代表了查询算子的出现次数、层次信息和查询子集，并从现有的基准查询模板中挑选出具有代表性的模板，将每个模板生成的查询作为目标查询.然后根据图映射的方式建模查询和目标查询的基本图模型，并通过计算两者之间的 GED 来减少计算量. 此外，该研究使用增量学习的方式进行特征选择，并验证了 k 近邻算法相比于支持向量回归算法更适合作为 SPARQL 基数估计的模型. 虽然该研究在减少计算量上做了一些改进，但固定特征数量会损失一定的估计准确性.

由于 SPARQL 中的三元组之间存在许多（自）连接，因此 SPARQL 查询估计不应该基于连接一致性假设[43-44]. 为了遵循这个规则，同时适用于 SPARQL 查询中的多种类型的查询模式[45-46]，例如星形或链式查询模式，Davitkova 等人[47] 提出了一个用于资源描述框架（resource description framework, RDF）图中基数估计的学习模型框架 LMKG. 具体来说，模型 LMKG 提出了一个子图编码方式 SG-Encoding，该编码方式实现了在稠密空间中表示各种子图模式，从而可以适用各种查询模式. 同时，LMKG 提供了一种无监督的自回归模型和一个基于深度神经网络的监督模型来进行基数估计，以适应不同的查询输入. 该研究通过学习子图模型之间的相关性而不是三元组之间的相关性，打破了连接一致性假设，同时该技术通过采样的方法来生成训练数据，缓解了处理大量数据耗时长的问题.

2.3 小 结

本文首先给出了查询驱动的基数估计的一般性

建模路线,详细分析了查询驱动的基数估计技术中相关的模型,这些模型使用了不同的编码方式,将查询负载编码为不同的特征,从而建立查询与数据库中列属性和表之间的联系,以支持不同的查询操作.同时,这些模型基于不同的监督学习方法,建立了查询及其基数之间的映射.此外,在包括高维图像、文本和复杂的 SPARQL 三元组的 NoSQL 查询中,查询驱动的基数估计模型也得到了很好的应用.本文对比了 2 类查询驱动的基数估计中的代表模型[3, 5, 9, 28],以及 NoSQL 代表模型[27,47],如表 2 所示.基于神经网络的基数估计技术相较于基于树结构的基数估计技术,由于神经网络优越的数据拟合能力,其估计准确性较高,但模型参数较大,导致运行时间较长.基于树结构的基数估计技术由于使用了增量学习,其模型参数相对较少,但以损失精度为代价.从表 2 中可以看到,查询驱动的基数估计模型的核心在于编码查询的特征是什么及使用哪种回归模型建模.查询驱动的基数估计方法在准确度上取得了良好的效果,使用查询特征编码器容易建模表之间、列之间的关系,并支持多表连接估计.

Table 2 Comparison of Query-Driven Cardinality Estimation Models

表 2 查询驱动的基数估计模型比较

模型	方法论	监督学习模型	支持的操作	输入	参数更新方法	查询编码
NN_{ST}[3]	回归模型/强化学习	神经神经网络	查询和连接	查询、基数	梯度更新	数据库属性信息、单一操作运算符
MSCN[5]	回归模型	卷积神经网络	查询和多表连接	查询、数据和基数	梯度更新	谓词、表和连接
Tree-based ensembles/NN[9]	回归模型	XGboost	查询	查询、数据和基数	梯度更新和贪心超参调整	谓词
XGBoost[28]	回归模型	XGboost	查询和多表连接	查询、数据	迭代更新	谓词和表的选择度
GL[27]	回归模型	深度神经网络	图片、文本相似性查询和连接	查询、数据、阈值和基数	梯度更新	查询、距离和阈值
LMKG[47]	回归模型/自回归模型	深度神经网络/深度自回归模型	SPARQL 查询	RDF 查询三元组、基数	梯度更新	查询的子图模式

3 数据驱动的基数估计技术

数据驱动的基数估计技术的核心是拟合数据库 D 中 N 维(即表的 N 列)数据的联合分布函数 $f_\theta(D)=P(A_1,A_2,\cdots,A_N)$,其中 θ 表示模型需要学习的参数,A_i 表示数据库表中的列属性.数据驱动的基数估计技术通常使用不同的采样策略,通过不同的监督/无监督的模型直接建模表中所有列的关系.首先,本文基于已有的数据驱动的基数估计技术归纳出数据驱动的基数估计的一般性建模流程;然后对不同数据驱动的基数估计模型进行了详细的介绍;最后对相关模型进行了对比和总结.

3.1 数据驱动的基数估计建模流程

数据驱动的基数估计技术的核心思想是建立一个数据驱动的基数估计模型,其一般性的建模流程如图 3 所示,分为 3 部分[23-24]:输入、输出和模型建立.

1)输入

输入数据库 D、查询 q.例如,对于数据驱动的代表模型 Naru[17],其模型的输入为数据库 D 和查询 q.其中,在模型推理阶段需要输入查询 q,而训练阶段仅需要给定数据库(表)中存储的数据.

2)输出

输出 N 维数据的联合概率分布 P 以及查询估计的选择度 $\hat{S}$.

3)模型建立

①模型训练.如图 3 中粗实线条箭头所示,数据驱动的基数估计模型包含 2 个模块:数据采样和数据分布模型.模型可选择不同的数据采样方法,并基于数据库中的数据采样合理规模的元组(多列数据).在这些高维数据的基础上,使用基于监督/无监督学习的数据分布模型拟合这些元组的分布,不断训练参数 θ 使得模型拟合最优的数据分布来作为整个数据库的数据分布.

本文以模型 Naru 为例来解释数据驱动的基数估计模型训练的流程.首先 Naru 扫描数据表,采样合理规模的元组作为数据集,其次,Naru 编码采样的每个元组作为自回归模型的输入向量,元组在每列样本上的数据分布作为其训练标签.对于每个元组,自回归模型以先前列的取值为输入,估计当前列的概率分布向量.通过训练缩小该概率分布向量与该列实际的数据分布之间的误差,从而得到一个可以处理所有列之间关系的自回归函数.

②模型推理.如图 3 中细实线条箭头所示,基数

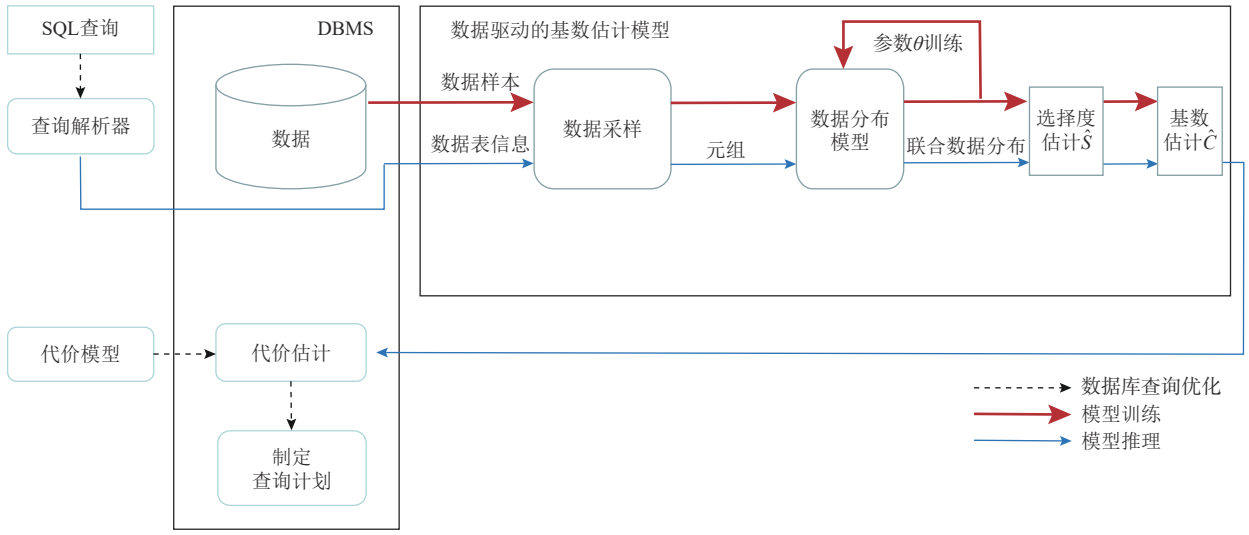


Fig. 3 General modeling flow for data-driven cardinality estimation

图 3 数据驱动的基数估计的一般性建模流程

估计在推理阶段同样需要经过 2 个模块. 不同的是, 数据分布模型已经确定, 根据查询 q, 经过查询解析器后, 根据数据表的信息(例如列的编码信息、统计信息等), 采样出若干条满足查询条件的元组. 将这些元组输入到数据分布模型中计算元组对应列的联合概率分布, 即为该查询 q 的选择度 $\hat{S}$, 从而计算出基数 $\hat{C}$.

同理以模型 Naru 为例来解释数据驱动的基数估计模型推理的流程. 对于等值查询 q_1, 首先 Naru 根据 q_1 和每列的数据分布进行采样生成元组, 并对这些元组进行编码. 对于范围查询 q_2, Naru 提出了一个渐进式采样的方式, 在已知数据分布的情况下对范围涵盖的样本进行采样, 并将采样出的元组转换为等值查询这种点查询. 将编码后的样本向量输入到训练好的自回归模型中产生 n 个条件概率, 并将这 n 个条件概率相乘即为查询 q 的选择度 $\hat{S}$, 从而计算出其基数 $\hat{C}$.

3.2 数据驱动的基数估计相关模型

数据驱动的基数估计模型的挑战是如何无需基于多列独立假设来建模单表多列之间的关系, 以及多表连接中表与表之间的依赖关系. 从是否使用监督学习的角度来看, Heimel 等人[15] 使用核函数进行监督学习, Park 等人[16] 使用均匀混合模型建模多列之间的关系; 文献 [17−20, 48−49] 使用无监督学习模型建模数据分布模型; 文献 [17−19] 基于自回归模型拟合数据分布; Halford 等人[48] 基于贝叶斯网络进行建模; Liu 等人[49] 提出了基于多维直方图的模型用于范围查询的基数估计. 此外, 目前大多数文献[15-18,48-49] 专注于建模多列之间的依赖关系, 而表之间的关系建模建立在多表独立假设之上. 最新的文献 [19] 基于自回归模型, 以及 Hilprecht 等人[20] 利用和积网络建模了表之间的关系, 很好地补充了查询驱动的基数估计模型中多表查询的相关工作. 因此, 本文将模型根据是否建模多列之间还是多表之间的关系, 分为面向多列连接查询的基数估计模型和面向多表连接查询的基数估计模型.

3.2.1 面向多列连接查询的基数估计模型

模型 Feedback-KDE[15] 将基于高斯核函数的密度估计器应用到基数估计技术中, 用于估计单表多列查询的选择度. 该模型首先从数据库表中随机采样合理规模的元组, 并基于高斯核函数初始化高斯密度模型的带宽. 接着使用训练集中的查询和其对应的真实基数训练高斯模型, 来优化高斯核的带宽. 因此, 模型 Feedback-KDE 是一个有监督训练的模型, 其为每种连接类型训练一个高斯模型. 当输入一个新的查询后, Feedback-KDE 首先解析该查询的连接类型, 并根据其连接类型找到对应的高斯模型.

模型 QuickSel[16] 使用均匀混合模型来表示数据的分布, 从而拟合给定查询的基数. 该模型支持范围查询, 其查询范围由查询中的谓词定义. 首先 QuickSel 从查询范围中随机采样数据, 并建立统一的模型. 然后计算真实查询范围和选择样本中的范围之间的重叠度并通过二次规划优化权值. 最后根据混合密度函数计算累计分布(选择度). 当输入一个新的查询后, QuickSel 首先解析该查询的连接类型, 然后根据其连接类型找到对应的混合模型.

作为自回归模型中具有代表性的模型，模型 Naru[17] 没有引入列的独立性这一假设，提出使用自回归模型学习查询 q 每列样本在前一列样本条件下的条件概率分布(选择度)，将所有列的选择度相乘得到查询 q 的选择度. 模型 DLM[18] 同样是基于自回归模型，与 Naru 不同的是，该模型将数据表中的一行数据看作一个序列，并将序列表示为由多个二进制向量组成. 同 Naru 一样，DLM 同样支持点查询和范围查询. 对于点查询操作，DLM 将查询中二进制向量的条件概率相乘来估计基数；对于范围查询，DLM 使用了重要性采样策略进行采样.

不同于基于自回归的模型，模型 Extended Chow-Liu trees[48] 从贝叶斯网络模型角度出发，将单表多列之间的依赖关系使用贝叶斯网络转换为条件概率. 该模型启发式地选择 Chow-Liu 树作为最优的贝叶斯网络结构，将高维数据联合分布转换为多个 2 维数据的条件概率分布. 此外，该模型提出了一个带有间隔的等高条件概率分布，用于减少存储条件概率分布的代价. 以等值查询为例，首先解析查询中包含的谓词；其次在构建好的 Chow-Liu 树中，根据涉及到的列抽取 Steiner 树；最后使用变量消除算法计算其选择度，从而得到基数. 同时，此模型同样适合单表的范围查询.

为了降低多维数据在查询估计中的查询成本，Liu 等人[49] 提出了以 RMI[50] 的思想将 MEH[51] 转换为可学习的多维直方图模型 LHist 来学习数据库的空间概要. LHist 为一个 d 层完全 K 叉树，其中 d 为数据的维度，K 是在每维的分割数. 树中的每个节点都为一个可训练的回归模型，用于预测输入此节点数据的索引，并输出该数据在下一层的节点的位置. 以范围查询为例，当输入一个新的范围查询后，LHist 首先编码其范围谓词，并将该查询转换为一个超矩形，然后通过 LHist 找到叶子节点，并计算范围谓词向量对应的索引值，最终将不同列上 2 个索引值的差加和得到查询的基数.

3.2.2 面向多表连接查询的基数估计模型

模型 NeuroCard[19] 可以看作模型 Naru 多表连接的扩展版本，该模型在所有列连接后的大表上通过使用加权抽样的方式均匀地选择样本并分批扫描大表，对于每批扫描得到的结果行，NeuroCard 首先编码每个属性值的嵌入表示输入到模型中. 其次模型更新自回归模型的参数并进行训练. 同 Naru 模型一样，NeuroCard 同样考虑了点查询和范围查询. 对于范围查询，同样使用了渐进式采样策略.

模型 DeepDB[20] 与模型 Naru 思想相似，同样是要学习所有列的联合概率分布. 不同的是，DeepDB 基于和积网络建模数据分布模型并支持多表连接. 该模型首先将数据库中所有的表连接为一个大表作为数据集，其次将该数据集中的样本按行聚类为若干个具有相似分布的数据组，然后将数据组中的样本按列分割成子节点. 对于同一组的叶子节点，使用一个乘积节点聚合其估计的选择度. 其次，使用加和节点组合成不同组的数据. DeepDB 通过拟合乘积节点中数据的联合分布概率，训练与加和节点之间边的权重，用于建模每个分组分布对整体联合分布的影响.

3.3 小 结

针对数据驱动的基数估计技术，本文首先给出了数据驱动的基数估计的一般性建模流程，其次分析了面向多列和多表连接查询的代表模型. 这些模型使用了不同的采样方式，将查询语句在数据表中相应的元组进行采样，输入到不同的数据分布模型中. 而不同的数据分布模型基于不同的学习模型，建模多维数据的联合分布概率，从而建立起查询、列和表之间的关联. 本文从多个角度对比了数据驱动的基数估计中的代表模型，如表 3 所示. 从表 3 中可以看到，数据驱动的基数估计模型的核心在于采用何种采样方式处理查询的元组，以及采用哪种数据分布模型建立多列、多表之间的关系. 相对于查询驱动的基数估计模型，数据驱动的基数估计模型拥有更小的训练开销，但在数据库删除和更新等操作中表现不佳.

4 混合模型的基数估计技术

查询驱动和数据驱动的基数估计技术各有利弊，而目前仅有少量研究[52] 将两者之中的优点结合起来，提出将查询驱动和数据驱动结合的混合模型的基数估计技术，并相互弥补这 2 类模型的一些缺点. 其核心思想是建立一个混合查询负载和数据分布的基数估计模型 $f_\theta(q,C,D)$，来同时处理和学习查询负载与数据样本中的信息，其中 θ 表示模型需要学习的参数. 本文给出了混合模型的一般性建模流程，同时对混合模型的基数估计技术中具体的混合模型进行了介绍.

混合模型的一般性建模流程包含了查询驱动和数据驱动的基数估计的建模流程，同样定义为输入、输出和模型建立 3 个部分，其一般的建模流程如图 4 所示.

1)输入

Table 3 Comparison of Data-Driven Cardinality Estimation Models

表 3 数据驱动的基数估计模型比较

模型	学习方式	方法论	模型	多列查询	多表查询	采样方式
Feedback-KDE[15]	监督学习	联合数据分布	基于高斯的核密度模型	√	×	随机采样
QuickSel[16]	监督学习	联合数据分布	均匀混合模型	√	×	基于查询的样本
Naru[17]	无监督学习	联合数据分布	自回归模型	√	×	渐进式采样
DLM[18]	无监督学习	联合数据分布	自回归模型	√	×	重要性采样
NeuroCard[19]	无监督学习	联合数据分布	自回归模型	√	√	渐进式采样
DeepDB[20]	无监督学习	联合数据分布	和积网络	√	√	没有使用
Extended Chow-Liu trees [48]	无监督学习	联合数据分布	贝叶斯网络	√	×	没有使用
LHist[49]	无监督学习	联合数据分布 +回归模型	多维直方图	√	×	没有使用

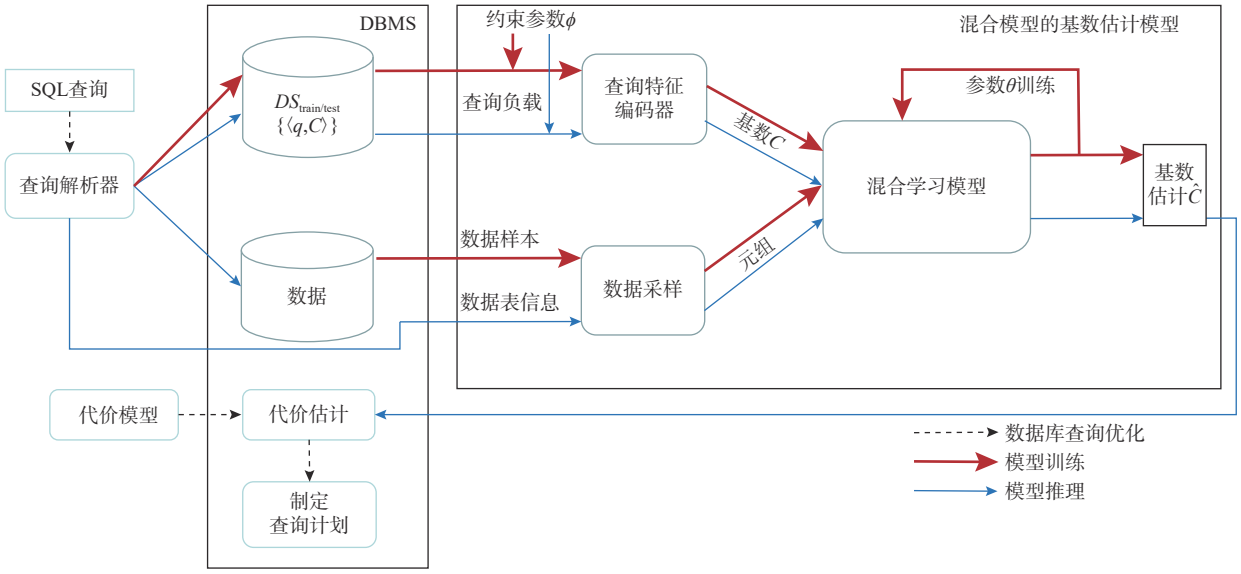


Fig. 4 General modeling flow for hybrid model cardinality estimation

图 4 混合模型基数估计的一般性建模流程

涵盖了查询驱动的输入和数据驱动的输入，即输入为数据集$DS_{\text{train/test}}\{\langle q,C\rangle\}$、模型约束参数$\phi$、数据库$D$和查询$q$.

2)输出

在数据库D中执行查询q估计的基数$\hat{C}$.

3)模型建立

①模型训练. 如图 4 中粗实线条箭头所示，混合分析模型共包含 3 个模块：查询特征编码器、数据采样和混合学习模型. 首先模型遵循查询驱动的基数估计中的建模流程，将输入的查询q进行编码. 同时遵循数据驱动的基数估计中的建模流程，将数据表中的数据采样出一定规模的元组. 其次，利用查询特征编码器和数据采样 2 个模块进一步加工查询负载和数据样本的信息. 具体来说，既可以通过将元组输入到查询特征编码器中进行元组编码后作为查询q特征属性的一部分，又可以通过查询q来采样更加精准的元组以拟合数据表中数据的分布. 将真实的基数和采样后的元组输入到混合学习模型中，使用监督学习和无监督学习的损失函数同时优化模型的学习参数θ.

UAE 模型[52]是第一个同时考虑查询驱动和数据驱动的模型，其既包含数据驱动的无监督学习，又拥有查询驱动的有监督学习的能力. 该模型创造性地在模型 Naru[17]的基础上融合了查询驱动的回归模型，共同学习数据分布概率参数. 本文以 UAE 模型为例来解释混合模型分析的训练流程. UAE 分为 2 个部分 UAE-D 和 UAE-Q. 其中 UAE-D 为数据驱动的基数估计模型 Naru，直接学习数据表中列数据之间的联合分布概率及得到数据分布在 Naru 的损失. 同时，将查询及其真实的选择度作为输入，首先使用与 Naru

同样的渐进式采样策略，将查询的元组进行采样，并使用 One-Hot 编码为向量；其次使用混合模型将不可微的采样元组向量转换为可微的元组向量输入到 Naru 的自回归模型中得到预测的选择度，并分析与真实选择度之间的误差和计算损失值. UAE 将 2 部分损失加和作为总损失，然后使用梯度更新的方式来优化自回归模型的参数.

②模型推理. 如图 4 中细实线条箭头所示，基数估计在推理阶段同样需要经过 3 个模块. 不同的是，混合学习模型的参数已经确定. 根据查询 q 可以得到其编码后的向量表示、采样的元组及其向量表示这 3 个元素，将这 3 个元素根据混合学习模型的设计，全部或部分输入到混合模型中直接估计出查询 q 的基数.

同样以 UAE 为例，UAE 将查询 q 采样的元组输入到自回归模型中得到估计的选择度，从而得到估计的基数. 这种混合模型的基数估计技术可以同时处理新增的数据以及具有新分布查询负载的情况. 对于增量数据，仅需要最小化 UAE-D 即对数据驱动的损失函数进行增量训练. 而 UAE 与训练查询负载相比，具有不同分布的查询，仅需要最小化 UAE-Q 即查询驱动的损失函数进行增量训练.

5 未来研究方向和展望

尽管基于机器学习的基数估计技术已经涌现出大量的研究，但其仍面临着 4 个挑战：

1)模型的训练成本和模型的可解释性. 针对查询驱动的基数估计模型，这类方法建模任务简单，占用空间小，仅使用简单的回归模型即可达到不错的效果. 然而，由于在收集训练数据时构建模型时间长，所以这类模型比较耗时，即使在文献 [28] 提出了方法来减轻开销，也以损失一定的精度为代价. 此外，由于建模技术基于监督学习，其模型的核心仍然是一个“黑盒”模型，因此模型的可解释性较弱. 因此，模型的训练成本和模型的可解释性是查询驱动的基数估计模型面临的两大重要挑战.

2)多表之间的复杂关系. 针对数据驱动的基数估计模型，这类方法相比于查询驱动的技术估计方法，其训练开销小，仅需要根据原始数据进行训练，且支持的查询范围更广泛. 虽然已有研究已经对多表之间的关系进行了建模并取得了一定的进展，但是这些研究均是直接将多个表连接为一个大表，进而转换为多列连接查询的思想进行建模，因此对于多表之间复杂关系的探索依然不足. 因此，多表之间的复杂关系是数据驱动的基数估计模型面临的一个重要挑战.

3)动态变化的数据库环境. 无论是查询驱动还是数据驱动的基数估计模型，目前的研究均不能适应动态变化的数据库环境[24]. 由于查询驱动的基数模型继承了监督学习的一些固有缺陷，使模型不适用于没有见过的查询，因此在频繁更新的数据库系统中该类模型估计的误差较大. 对于数据驱动的基数估计模型，由于其直接拟合数据库中的数据分布，因此该类模型可以支持新的查询. 但是，这类方法仍然不能支持数据的删除和更新等动态操作. 因此，支持动态变化的数据库环境是数据驱动和查询驱动的基数估计模型面临的一个共同挑战.

4)基数估计 benchmark 和评价体系. 目前，对已有方法进行评价的文献 [23-24] 仍然不丰富，所以设计 benchmark 对已有方法进行对比和评价也是基数估计领域中研究的一个重要挑战.

针对以上分析，基于机器学习的基数估计技术的未来研究方向可以包含 4 个方面：

1)考虑使用可解释的增量学习方法. 如何降低学习模型的成本和如何提高模型的可解释性是查询驱动的基数估计技术研究的两大方向. 文献 [28] 证明了增量学习可以降低查询驱动的基数估计模型的学习成本，但其估计准确性有所下降. 因此，探索可以提升估计精度的增量学习模型是降低模型学习成本有效且直观的解决方式. 随着可解释模型在人工智能领域的发展，机器学习模型逐渐从“黑盒”走向“透明”. 因此，将增量学习与可解释性模型相结合有助于探索一个全新的基于增量学习且拥有可解释性的查询驱动的基数估计模型.

2)建模多表之间的复杂关系. 针对于在数据驱动的基数估计技术中仍然缺乏探索的多表复杂关系，应当考虑设计可以建模多表之间关系的数据分布模型，以更好地估计多表连接查询的基数.

3)探索更加优越的混合模型. 混合查询驱动和数据驱动的基数估计技术可以结合两者优点，互补两者的缺点，文献 [52] 证明了混合模型的估计效果优于独立的查询驱动的基数估计模型和数据驱动的基数估计模型，但是混合模型目前成果较少，因此结合哪些机器学习模型或者数据结构，如何结合查询驱动的基数估计模型和数据驱动的基数估计模型也是未来研究的热点.

4)加快机器学习模型训练速度应部署到实际数

据库进行应用. 由于基于机器学习的模型更新速度较慢，无法适应频繁更新的动态数据库环境，距离实际落地应用还有一定的差距[24]. 因此，基于机器学习的基数估计技术作为一个重要的研究方向应以实际落地应用为目标进行设计，探索可以加快机器学习模型训练速度的算法，以减少模型参数调优的时间和提高模型查询效率，最终适应动态变化的数据库环境.

6 总　　结

本文首先介绍了基于机器学习的基数估计技术的相关背景；接着介绍了基数估计的相关定义、特征编码技术以及基数估计技术中的分类体系；然后分别从查询驱动、数据驱动和混合模型的角度出发，给出了3类基数估计的一般性建模流程；并基于建模流程，分别讨论了查询驱动、数据驱动和混合模型这3类基于机器学习的基数估计的代表性模型，并总结了其利弊；最后，本文还讨论了基于机器学习的基数估计技术中存在的挑战和未来的研究方向. 基于机器学习的基数估计技术近年来涌现了大量的研究，但距离落地到实际动态变化的数据库环境还有一定的差距，未来会有更多的方法和研究成果出现，在数据库实际应用中占有一席之地.

作者贡献声明：岳文静负责论文主要内容的调研和撰写，以及论文的校验；屈稳稳辅助调研相关内容并提供指导意见；林宽、王晓玲对文章的结构和内容提供指导意见.

参　考　文　献

[1] Lakshmi S, Zhou Shaoyu. Selectivity estimation in extensible databases−A neural network approach[C] //Proc of the 24th Int Conf on Very Large Data Bases. San Francisco, CA: Morgan Kaufmann, 1998: 623−627

[2] Lu Hongju, Setiono R. Effective query size estimation using neural networks[J]. Applied Intelligence, 2002, 16(3): 173−183

[3] Ortiz J, Balazinska M, Gehrke J, et al. Learning state representations for query optimization with deep reinforcement learning[C/OL] // Proc of the 2nd Workshop on Data Management for End-to-End Machine Learning. New York: ACM. 2018[2023-03-08]. https://dl.acm.org/doi/pdf/10.1145/3209889.3209890

[4] Woltmann L, Hartmann C, Thiele M, et al. Cardinality estimation with local deep learning models[C/OL] //Proc of the 2nd Int Workshop on Exploiting Artificial Intelligence Techniques for Data Management. New York: ACM, 2019 [2023-03-08]. https://dl.acm.org/doi/10.1145/3329859.3329875

[5] Kipf A, Kipf T, Radke B, et al. Learned cardinalities: Estimating correlated joins with deep learning[C/OL] //Proc of the 9th Biennial Conf on Innovative Data Systems Research. 2019[2023-03-08]. https://www.cidrdb.org/cidr2019/papers/p101-kipf-cidr19.pdf

[6] Ortiz J, Balazinska M, Gehrke J, et al. An empirical analysis of deep learning for cardinality estimation[J]. arXiv preprint, arXiv: 1905.06425, 2019

[7] Malik T, Burns R C, Chawla N V. A black-box approach to query cardinality estimation[C/OL] //Proc of the 3rd Biennial Conf on Innovative Data Systems Research. 2007[2023-03-08]. https://www.cidrdb.org/cidr2007/papers/cidr07p06.pdf

[8] Sun Ji, Li Guoliang. An end-to-end learning-based cost estimator[J]. Proceedings of the VLDB Endowment, 2019, 13(3): 307−319

[9] Dutt A, Wang Chi, Nazi A, et al. Selectivity estimation for range predicates using lightweight models[J]. Proceedings of the VLDB Endowment, 2019, 12(9): 1044−1057

[10] Marcus R, Papaemmanouil O. Towards a hands-free query optimizer through deep learning[J]. arXiv preprint, arXiv: 1809. 10212, 2018

[11] Liu H, Xu Mingbin, Yu Ziting, et al. Cardinality estimation using neural networks[C] //Proc of the 25th Annual Int Conf on Computer Science and Software Engineering. New York: ACM, 2019: 53−59

[12] Liu Jie, Dong Wenqian, Li Dong, et al. Fauce: Fast and accurate deep ensembles with uncertainty for cardinality estimation[J]. Proceedings of the VLDB Endowment, 2021, 14(11): 1950−1963

[13] Marcus R, Kipf A, Van Renen A, et al. Flow-loss: Learning cardinality estimates that matter[J]. Proceedings of the VLDB Endowment, 2021, 14(11): 2019−2032

[14] Tzoumas K, Deshpande A, Jensen C S. Lightweight graphical models for selectivity estimation without independence assumptions[J]. Proceedings of the VLDB Endowment, 2011, 4(11): 852−863

[15] Heimel M, Kiefer M, Markl V. Self-tuning, GPU-accelerated kernel density models for multidimensional selectivity estimation[C] //Proc of the 2015 ACM SIGMOD Int Conf on Management of Data. New York: ACM, 2015: 1477−1492

[16] Park Y, Zhong Shucheng, Mozafari B. QuickSel: Quick selectivity learning with mixture models[C] //Proc of the 2020 ACM SIGMOD Int Conf on Management of Data. New York: ACM, 2020: 1017−1033

[17] Yang Zongheng, Liang E, Kamsetty A, et al. Deep unsupervised cardinality estimation[J]. Proceedings of the VLDB Endowment, 2019, 13(3): 279−292

[18] Hasan S, Thirumuruganathan S, Augustine J, et al. Deep learning models for selectivity estimation of multi-attribute queries[C] //Proc of the 2020 ACM SIGMOD Int Conf on Management of Data. New York: ACM, 2020: 1035−1050

[19] Yang Zongheng, Kamsetty A, Luan Sifei, et al. NeuroCard: One cardinality estimator for all tables[J]. Proceedings of the VLDB Endowment, 2021, 14(1): 61−73

[20] Hilprecht B, Schmidt A, Kulessa M, et al. DeepDB: Learn from data, not from queries![J]. Proceedings of the VLDB Endowment, 2020, 13(7): 992−1005

[21] Zhu Rong, Wu Ziniu, Han Yuxing, et al. FLAT: Fast, lightweight and accurate method for cardinality estimation[J]. arXiv preprint, arXiv: 2011. 09022, 2020

[22] Lan Hai, Bao Zhifeng, Peng Yuwei. A survey on advancing the dbms query optimizer: Cardinality estimation, cost model, and plan enumeration[J]. Data Science and Engineering, 2021, 6(1): 86−101

[23] Sun Ji, Zhang Jintao, Sun Zhaoyan, et al. Learned cardinality estimation: A design space exploration and a comparative evaluation[J]. Proceedings of the VLDB Endowment, 2021, 15(1): 85−97

[24] Wang Xiaoying, Qu Changbo, Wu Weiyuan, et al. Are we ready for learned cardinality estimation?[J]. Proceedings of the VLDB Endowment, 2021, 14(9): 1640−1654

[25] Li Guoliang, Zhou Xuanhe, Sun Ji, et al. Survey on database technology based on machine learning[J]. Chinese Journal of Computers, 2020, 43(11): 2019−2049 (in Chinese)
(李国良, 周煊赫, 孙佶, 等. 基于机器学习的数据库技术综述[J]. 计算机学报, 2020, 43(11): 2019−2049)

[26] Leis V, Gubichev A, Mirchev A, et al. How good are query optimizers, really?[J]. Proceedings of the VLDB Endowment, 2015, 9(3): 204−215

[27] Sun Ji, Li Guoliang, Tang Nan. Learned cardinality estimation for similarity queries[C] // Proc of the 2021 Int Conf on Management of Data. New York: ACM, 2021: 1745−1757

[28] Dutt A, Wang Chi, Narasayya V, et al. Efficiently approximating selectivity functions using low overhead regression models[J]. Proceedings of the VLDB Endowment, 2020, 13(11): 2215−2228

[29] Acharya J, Diakonikolas I, Hegde C, et al. Fast and near-optimal algorithms for approximating distributions by histograms[C] //Proc of the 34th ACM SIGMOD-SIGACT-SIGAI Symp on Principles of Database Systems. New York: ACM, 2015: 249−263

[30] Xu Jia, Zhang Zhenjie, Xiao Xiaokui, et al. Differentially private histogram publication[J]. The VLDB Journal, 2013, 22(6): 797−822

[31] Ioannidis Y. The history of histograms (abridged)[C] //Proc of the 29th Int Conf on Very Large Data Bases. San Francisco, CA: Morgan Kaufmann, 2003: 19−30

[32] Tschorsch F, Scheuermann B. An algorithm for privacy-preserving distributed user statistics[J]. Computer Networks, 2013, 57(14): 2775−2787

[33] Flajolet P, Éric F, Gandouet O, et al. HyperLogLog: The analysis of a near-optimal cardinality estimation algorithm[J]. Analysis of Algorithms , 2007: 137−156

[34] Alon N, Matias Y, Szegedy M. The space complexity of approximating the frequency moments[J]. Journal of Computer and System Sciences, 1999, 58(1): 137−147

[35] Durand M, Flajolet P. Loglog counting of large cardinalities[C] //Proc of the 11th Annual European Symp. Berlin: Springer, 2003: 605−617

[36] Wu Wentao, Naughton J F, Singh H. Sampling-based query re-optimization[C] //Proc of the 2016 Int Conf on Management of Data. New York: ACM, 2016: 1721−1736

[37] Chen Yun, Yi Ke. Two-level sampling for join size estimation[C] // Proc of the 2017 ACM Int Conf on Management of Data. New York: ACM, 2017: 759−774

[38] Leis V, Radke B, Gubichev A, et al. Cardinality estimation done right: Index-based join sampling[C/OL] //Proc of the 8th Biennial Conf on Innovative Data Systems Research. 2017[2023-03-08].https://www.cidrdb.org/cidr2017/papers/p9-leis-cidr17.pdf

[39] Lipton R J, Naughton J F, Schneider D A. Practical selectivity estimation through adaptive sampling[C] //Proc of the 1990 ACM SIGMOD Int Conf on Management of Data. New York: ACM, 1990: 1−11

[40] Olken F, Rotem D. Random sampling from database files: A survey[C] //Proc of the 5th Int Conf on Scientific and Statistical Database Management. Berlin: Springer, 1990: 92−111

[41] Hasan R. Predicting sparql query performance and explaining linked data[C] //Proc of the 11th Int Conf on Semantic Web: Trends and Challenges. Berlin: Springer, 2014: 795−805

[42] Zhang W E, Sheng Q Z, Qin Yongrui, et al. Learning-based SPARQL query performance modeling and prediction[J]. World Wide Web, 2018, 21(4): 1015−1035

[43] Neumann T, Weikum G. The RDF-3X engine for scalable management of RDF data[J]. The VLDB Journal, 2010, 19(1): 91−113

[44] Stocker M, Seaborne A, Bernstein A, et al. SPARQL basic graph pattern optimization using selectivity estimation[C] //Proc of the 17th Int Conf on World Wide Web. New York: ACM, 2008: 595−604

[45] Vidal M E, Ruckhaus E, Lampo T, et al. Efficiently joining group patterns in SPARQL queries[C] //Proc of the 7th Extended Semantic Web Conf. Berlin: Springer, 2010: 228−242

[46] Huang Hai, Liu Chengfei. Estimating selectivity for joined RDF triple patterns[C] //Proc of the 20th ACM Int Conf on Information and Knowledge Management. New York: ACM, 2011: 1435−1444

[47] Davitkova A, Gjurovski D, Michel S. LMKG: Learned models for cardinality estimation in knowledge Graphs[J]. arXiv preprint, arXiv: 2102. 10588, 2021

[48] Halford M, Saint-Pierre P, Morvan F. An approach based on Bayesian networks for query selectivity estimation[C] // Proc of the 24th Int Conf on Database Systems for Advanced Applications. Berlin: Springer, 2019: 3−19

[49] Liu Qiyu, Shen Yanyan, Chen Lei. LHist: Towards learning multi-dimensional histogram for massive spatial data[C] // Proc of the 37th IEEE Int Conf on Data Engineering. Piscataway, NJ: IEEE, 2021: 1188−1199

[50] Kraska T, Beutel A, Chi E H, et al. The case for learned index structures[C] //Proc of the 2018 Int Conf on Management of Data. New York: ACM, 2018: 489−504

[51] Muralikrishna M, Dewitt D J. Equi-depth histograms for estimating selectivity factors for multi-dimensional[C] //Proc of the 1988 ACM SIGMOD Int Conf on Management of Data. New York: ACM, 1988: 28−36

[52] Wu Peizhi, Cong Gao. A unified deep model of learning from both data and queries for cardinality estimation[C] //Proc of the 2021 Int Conf on Management of Data. New York: ACM, 2021: 2009−2022



Yue Wenjing, born in 1996. PhD candidate. Student member of CCF. Her main research interests include database, cardinality estimation techniques, machine learning, and recommender system.

岳文静, 1996 年生. 博士研究生. CCF 学生会员. 主要研究方向为数据库、基数估计技术、机器学习、推荐系统.



Qu Wenwen, born in 1995. PhD candidate. His main research interests include database, graph processing, and distributed graph mining system. (wenwenqu@stu.ecnu.edu.cn)

屈稳稳, 1995 年生. 博士研究生. 主要研究方向为数据库、图计算、分布式图挖掘系统.



Lin Kuan, born in 1982. Master, associate professor. Her main research interest includes signal and information processing. (linkuan@aircas.ac.cn)

林　宽, 1982 年生. 硕士, 副研究员. 主要研究方向为信号与信息处理.



Wang Xiaoling, born in 1975. PhD, professor, PhD supervisor. Senior member of CCF. Her main research interests include knowledge graph, personalized recommendation, and privacy protection.

王晓玲, 1975 年生. 博士, 教授, 博士生导师. CCF 高级会员. 主要研究方向为知识图谱、个性化推荐、隐私保护.