基于查询特征表示学习的联邦复杂查询基数估计

徐 娇[1,2,3,4],田萍芳[1,2,3,4],顾进广[1,2,3,4],徐芳芳[1,2,3,4]
(1. 武汉科技大学 计算机科学与技术学院,湖北 武汉 430065;
2. 湖北省智能信息处理与实时工业系统重点实验室,湖北 武汉 430065;
3. 武汉科技大学 大数据科学与工程研究院,湖北 武汉 430065;
4. 国家新闻出版署富媒体数字出版内容组织与知识服务重点实验室,北京 100083)

摘 要:准确的基数估计是实现最佳查询计划的关键因素,现有方法大多基于深度学习来解决基数估计问题。然而,这种基于 RDF 图模式的方法专注于具有特定拓扑结构的简单查询,适用范围有限,缺乏对现实场景中频繁使用的复杂类查询的支持。为了解决以上问题,提出一种基于查询特征表示学习的联邦复杂查询基数估计模型。该模型主要处理带有 FILTER 或 DISTINCT 关键字的复杂查询,使用新提出的 FILTER 查询特征化方法将 SPARQL 查询表示为特征向量,通过模型预测查询基数。同时使用模型预测 DISITINCT 查询中唯一行比率。在 LUBM 数据集上的实验表明,与最先进的基数估计方法相比,该模型在估计质量上表现优异,平均估计误差中位数可达 1.16,并对多连接查询的基数估计表现出潜力和可扩展性。

关键词:联邦系统;查询优化;复杂查询;深度学习;基数估计
中图分类号:TP319 **文献标识码**:A **文章编号**:1673-629X(2024)02-0032-08
doi:10.3969/j.issn.1673-629X.2024.02.005

Cardinality Estimation of Federated Complex Queries Based on Query Feature Representation Learning

XU Jiao[1,2,3,4],TIAN Ping-fang[1,2,3,4],GU Jin-guang[1,2,3,4],XU Fang-fang[1,2,3,4]
(1. School of Computer Science and Technology,Wuhan University of Science and Technology,Wuhan 430065,China;
2. Hubei Province Key Laboratory of Intelligent Information Processing and Real-time Industrial System,
Wuhan 430065,China;
3. Institute of Big Data Science and Engineering Research,Wuhan University of Science and Technology,
Wuhan 430065,China;
4. Key Laboratory of Rich Media Digital Publishing Content Organization and Knowledge Service,
National Press and Publication Administration,Beijing 100083,China)

Abstract:Accurate cardinality estimation is the key factor to realize the best query plan. Most of the existing methods are based on deep learning to solve the base estimation problem. However,this method based on RDF graph pattern focuses on simple queries with specific topological structure,which is limited in application scope,and lacks support for complex queries frequently used in real scenes. In order to solve the above problems,we propose a federated complex query cardinality estimation model based on query feature representation learning. This model mainly deals with complex queries with FILTER or DISTINCT keywords. The SPARQL query is expressed as a feature vector by using the newly proposed FILTER query characterization method,and the query cardinality is predicted by the model. Also the model is used to predict the ratio of unique rows in DISITINCT queries. Experiments on LUBM data sets show that compared with the most advanced cardinality estimation methods,this model performs better in cardinality estimation,with an average median estimation error of 1.16,and shows potential and scalability for the estimation of multi-join queries.

收稿日期:2023-03-30 **修回日期**:2023-07-31
基金项目:科技创新 2030“新一代人工智能”重大项目(2020AAA0108500);国家自然科学基金(U1836118);富媒体数字出版内容组织与知识服务重点实验室开放基金(ZD2021-11/01)
作者简介:徐 娇(1994-),女,硕士,通讯作者,研究方向为联邦查询;田萍芳(1972-),女,博士,教授,CCF 会员(E4268M),研究方向为计算机网络、语义网等;顾进广(1974-),男,博士,教授,CCF 杰出会员(05460D),研究方向为语义网与知识图谱、分布式计算;徐芳芳(1988-),女,硕士,实验师,CCF 会员(56085M),研究方向为语义数据管理、数据查询优化。

Key words: federal system; query optimization; complex query; deep learning; cardinality estimation

0　引　言

准确的基数估计是实现最佳查询计划的关键因素。现有方法大多基于深度学习来解决基数估计问题,例如,在给定知识图谱的情况下,Davitkova A 等人[1]提出了 LMKG 方法来学习和估计最常用的查询类型(即星形和链形查询)的基数,通过将图模式的基数估计问题表示为一个深度学习任务,有效地捕获不同子图模式之间的相关性,从而提供非常准确的估计结果。然而,这种基于 RDF 图模式的方法专注于具有特定拓扑结构的简单查询,适用范围有限[2-3]。例如,在现实场景中,带有 FILTER 或 DISTINCT 运算符的复杂 SPARQL 语句构成了一大类频繁使用的查询[4],但现有方法缺乏对这类复杂查询的基数估计支持。

为了解决以上问题,该文提出基于查询特征表示学习的联邦复杂查询基数估计模型,以学习和预测输入查询的基数。模型包含两个方法,基于全连接多层神经网络(Multi-Layer Perceptron,MLP)来预测结果。对于 FILTER 类查询,将查询表示为:顶点集、边集、连接集和 FILTER 条件子句集四个特征,FILTER 条件子句集使用新提出的特征化方法编码,其他三个特征集则使用 SG-Encoding 进行编码,将合并的向量集作为参数输入模型,模型预测查询基数。而对于 DISTINCT 查询,使用模型预测唯一行的比率。结果表明,该方法能得到更精确的基数估计,具有实际应用价值。

综上所述,该文的贡献可以归纳为:

(1)提出了一种 FILTER 查询特征化的方法,分为简单范围谓词编码和通用合取编码,将编码得到的特征向量作为模型的输入,解决了 SPARQL 联邦查询向量化问题;

(2)提出了一种基于 MLP 估计唯一率的方法,以实现 DISTINCT 查询估计不包含重复项基数的功能;

(3)提出了基于查询特征表示学习的联邦复杂查询基数估计模型,该模型可以学习并预测联邦系统中包含 FILTER 或 DISTINCT 关键字的 SPARQL 查询。

1　相关工作

以往的研究表明,在 WHERE 子句中包含 AND,OR 和 NOT 运算符的查询构成了一大类频繁使用的查询,它们的表达能力大致相当于关系代数,而在 SPARQL 系统中,这类查询通常由 FILTER 运算符进行标识和连接。此外,对于具有 DISTINCT 以及在计划中的查询,查询规划器需要集合论基数,例如,在考虑排序选项时。因此,上述两类查询的基数估计对于查询优化非常重要[5-7]。另外,对于深度学习的基数估计而言,查询特征化技术是必要的[8]。

深度学习用于基数估计在 SQL 领域已进行了深入研究[9-17]。MSCN[7]模型基于神经网络来支持具有多个谓词的基表和连接大小估计,但其查询特征化缺乏领域知识和可解释性,因为在训练过程中通过其结构学习隐含的黑盒特征。Naru[18]使用自回归模型来学习点查询的条件联合概率,但会增加范围查询的开销,因为它们的估计是多个点查询的总和。DeepDB[19]则在一定程度上依赖于采样来寻找匹配的连接属性构建 SPN。基于树型门控循环单元的方法[20]同时对基数和代价进行估计,能够有效学习计划与基数和代价之间的高维关系,进而给出精确的估计结果。

在 SPARQL 联邦查询中,SPLENDID[21]使用 VOID 统计信息和基于成本的基数估计模型为联邦查询选择执行计划,但其成本模型没有涵盖分组、聚合和 SERVICE 子查询等复杂查询场景。Odyssey[22]基于特征集方法,在基数估计时考虑了使用 DISTINCT 修饰符的查询,但所使用的共享相同属性集的实体相似原则主要适用于星形查询。CostFed[23]基于数据摘要文件来估计查询成本,其中通过创建资源桶来考虑资源频率分布的不对称性,以至于对数据集中高频三元组模式估计质量好,但对于低频三元组模式则表现较差。基于查询特征表示学习的联邦知识查询基数估计方法[24]通过将 SPARQL 查询表示为特征向量,使用 CEFQR 模型学习和预测查询中的基数。虽然该模型在基数估计问题上表现优异,但是如上文所述,该方法缺乏对复杂查询基数估计的支持。

受到 SQL 领域的启发,笔者认为可以将 SPARQL 复杂联邦查询的基数估计问题表示为一个监督学习任务,标签是实际基数,输入的是查询特征,输出的是预测的基数。相较于查询特征表示学习的方法(CEFQR[24]),文中方法具有以下创新:首先,文中方法不再局限于简单联邦查询,而是将模型扩展为支持复杂查询的基数估计。相应地,提出了复杂类查询的编码技术;其次,文中模型除了预测 SPARQL 查询的基数外,还能估计查询中不重复结果的基数,应用范围更广泛。总得来说,文中模型更具实用价值。

2　基于查询特征表示学习的联邦复杂查询基数估计模型

2.1　模型概述

模型的整体架构如图 1 所示。根据输入查询类型

的不同,模型输出相应的预测值。当输入类型是包含 FILTER 关键字的查询时,模型输出预测基数 W_{out} ,相反,若为 DISTINCT 类查询则输出唯一率 R_{out} 。从输入查询到模型输出预测结果主要经历三个阶段:第一阶段,将输入查询转换成一组向量 $\boldsymbol{V}$, $\boldsymbol{V}$ 由多个集合组成。对于 FILTER 类查询, $\boldsymbol{V}=(\boldsymbol{A},\boldsymbol{X},\boldsymbol{E},\boldsymbol{F})$,其中 $\boldsymbol{A}$ 表示邻接张量, $\boldsymbol{X}$ 表示节点特征矩阵, $\boldsymbol{E}$ 表示谓词特征矩阵, $\boldsymbol{F}$ 表示 FILTER 条件子句特征矩阵;而针对 DISTINCT 类查询, $\boldsymbol{V}=(\boldsymbol{A},\boldsymbol{X},\boldsymbol{E})$ 关于两类查询的编码方式, $\boldsymbol{A},\boldsymbol{X},\boldsymbol{E}$ 矩阵的特征化表示使用 LMKG 所提出的 SG-Encoding 编码, $\boldsymbol{F}$ 条件子句的编码方式将在 2.2 节详细介绍。第二阶段,给定向量集合 $\boldsymbol{V}$,将 $\boldsymbol{V}$ 的每个向量作为 MLP_{out} 的输入, MLP_{mid} 是全连接的单层神经网络。然后 MLP_{mid} 将 $\boldsymbol{V}$ 中的每个向量集合连接合并成 H 维向量 $\boldsymbol{Q}_{vec}$,其中 $\boldsymbol{Q}_{vec}$ 表示 $\boldsymbol{V}$ 中所有元素的单个转换表示的平均值,即:

$$\boldsymbol{Q}_{vec}=\frac{1}{|V|}\sum_{v\in V}\mathrm{MLP}_{mid}(v) \tag{1}$$

$$\mathrm{MLP}_{mid}(v)=\mathrm{ReLU}(vU_{mid}+b_{mid}) \tag{2}$$

其中, $U_{mid}\in R^{LxH}$, $b_{mid}\in R^{H}$ 表示学习的权重和偏差,而 $v\in R^{L}$ 是输入行向量。选择一个平均值(而不是求和)来简化对集合 $\boldsymbol{V}$ 中不同数量元素的泛化。在第三阶段,使用两层神经网络 MLP_{out} 估计查询的预测基数 W_{out} 或唯一率 R_{out} ,对于 FILTER 类查询, $W_{out}=\mathrm{MLP}_{out1}(Q_{vec})$ 。唯一率 R_{out} 的计算过程将在 2.3 节详细讨论。

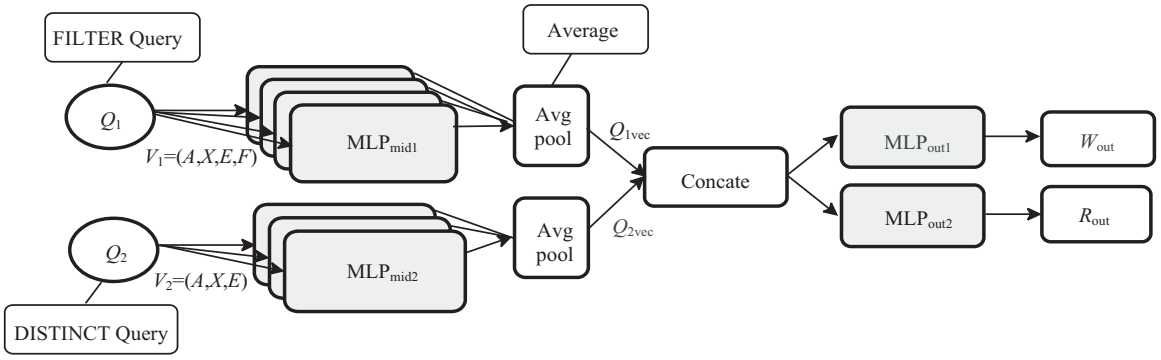

图 1 模型架构

模型对目标基数 C 进行归一化:首先取对数使目标值分布更均匀,然后使用从训练集获得的对数化后的最小值和最大值来归一化到区间[0,1]。归一化是可逆的,因此模型可以对预测结果 $W_{out}\in[0,1]$ 反归一化得到预测基数。

模型的构建包括两个步骤。首先,生成一个随机训练集。其次,使用训练集训练该模型,直到平均 Q-error 开始收敛到其最佳绝对值,Q-error 被定义为估计值(Y)与真实值(y)之间的比率,如公式 3。在训练阶段,使用了早停技术。此外,实验使用 TensorFlow[25] 框架和 Adam[26] 训练优化器来训练和测试模型。

$$Q-\mathrm{error}(y,Y)=\max\left(\frac{y}{Y},\frac{Y}{y}\right) \tag{3}$$

2.2 FILTER 查询特征化方法

如上所示,该文重点关注对 FILTER 条件子句的编码。

(1)简单范围谓词编码。

对于匹配单个变量的范围谓词查询,定义条件子句 $F=(\mathrm{var},\mathrm{op},\mathrm{val})$, var 表示用于筛选的变量名,在使用 SG-Encoding 对 (A,X,E) 进行编码时,会对子图节点(主语和宾语)和谓语进行排序,之后创建 Term-ID 映射列表, var 根据此列表进行 One-Hot 编码。op 表示比较运算符>, =, <中的任意一种,使用长度为 3 的二进制编码。val 表示比较的文字值,使用公式 4 将 var 归一化为[0,1]范围的 val^{*} 。

$$\mathrm{val}^{*}=\frac{\mathrm{val}-\min(\mathrm{var})}{\max(\mathrm{var})-\min(\mathrm{var})} \tag{4}$$

在进行范围谓词编码时,考虑了值的离散分布,将所有类型的点和范围谓词都编码到封闭区间。例如, $?x=5$ 变成[5,5], $?x\leqslant 5$ 变成[Min(x),5]。当条件子句的开放范围很大时,只需要对满足筛选条件且在变量值域范围内的值进行编码,来减少特征化过程的时间,这对训练模型是很有益的。例如图 2, 在 Q_1 查询中,给定 max(?age) = 50, min(?age) = 15, 对 FILTER(?age<24)子句的编码过程为:首先使用 Term-ID 映射列表将?age 编码为[10],然后根据变量的值域范围将筛选谓词限定为 15<?age<24,最后对下界 15 和上界 24 进行归一化,串联得到的向量 F 如图 2 所示。

(2)通用合取编码。

对于具有多个变量且每个变量存在多个谓词的范围查询,使用通用合取编码。通过观察发现,当条件子句中存在多个变量,且每个变量具有多个谓词时,变量必然属于主语、谓语或宾语中的一类,那么满足条件的值则一定存在于对应节点的值域范围内。因此,编码

步骤简述为:(a)对每类节点的数据域进行分区;(b)在特征向量中给每个分区一个条目;(c)给每个条目分配一个值,指示它所代表的分区是否满足查询 Q 中的谓词,使用 0 表示没有值满足条件,1/2 表示部分满足,1 表示都满足。其中,每类节点 $N(N \in \{s,p,o\})$ 的最大分区数为 $n(N) = \min(n, \max(N) - \min(N) + 1)$,$n$ 表示设定的最大分区数阈值。另外,特征向量中的条目 $v(v \in N)$ 具有基于零的索引 $\mathrm{index}(v)$,计算方式如式 5。

$$\mathrm{index}(v) = \frac{\mathrm{val} - \min(N)}{\max(N) - \min(N)} \cdot n_A \tag{5}$$

最后,每类节点特征化的连接产生总的特征向量 $\boldsymbol{F}$。例如图 3,给定查询 Q_2 中涉及的主语和宾语的最值为:$\min(S) = -9$,$\max(S) = 50$,$\min(O) = 0$,$\max(O) = 115$,并且 $n = 12$。对 FILTER (? id<7&& ? age >10 &&? age<30) 子句的编码过程为:首先,对于? id< 7,因为? id 属于主语,所以计算它在 S 分区中的索引,根据公式 5,index(? id) = 3,则在 S 分区中第四个条目设置为 1/2(索引从 0 开始),左侧的所有条目均为 1,表示小于 7 的值符合条件,相应地,右边的所有条目均设置为 0。同理,按照上述步骤处理? age 上的条件。最后得到向量 $\boldsymbol{F}$ 如图 3 所示。

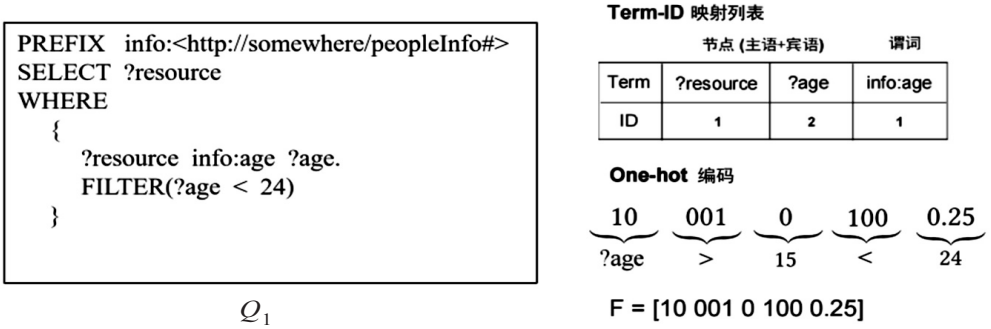

图 2　简单范围谓词编码过程

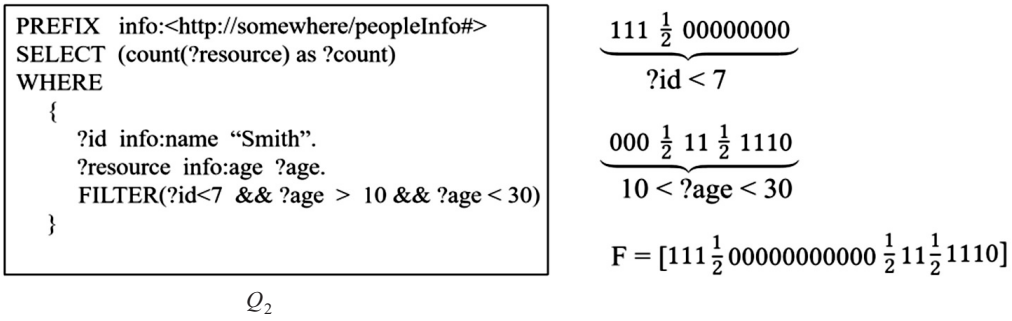

图 3　通用合取编码过程

2.3　估计唯一率

对于 DISTINCT 类查询,首先,给出唯一率的定义:如果 SPARQL 查询 Q 在 RDF 数据集 D 上的执行结果行(包含重复)中有 $x\%$ 是唯一的,那么查询 Q 在数据集 D 上的唯一率等于 $x\%$,计算公式为:

$$x\% = \frac{\| Q_D \|}{| Q_D |} \tag{6}$$

其中,Q_D 表示 Q 在 D 上的基数,运算符 ‖ ‖ 返回去除重复的基数,| | 返回包含重复项的基数。

使用完全连接的双层神经网络 $\mathrm{MLP}_{\mathrm{out}}$ 来计算输入查询的预测唯一率 R_{out}。首先,$\mathrm{MLP}_{\mathrm{out}}$ 将大小为 H 的 $\boldsymbol{Q}_{\mathrm{vec}}$ 向量作为输入,然后使用第一层将输入向量转换为大小为 $0.5H$ 向量,最后使用第二层将 $0.5H$ 向量转换为表示唯一率的单个值 R_{out},计算方式如下所示:

$$R_{\mathrm{out}} = \mathrm{MLP}_{\mathrm{out2}}(\boldsymbol{Q}_{\mathrm{vec2}}) \tag{7}$$

$$\mathrm{MLP}_{\mathrm{out2}}(v) = \mathrm{Sigmoid}(\mathrm{ReLU}(vU_{\mathrm{out1}} + b_{\mathrm{out1}})U_{\mathrm{out2}} + b_{\mathrm{out2}}) \tag{8}$$

其中,R_{out} 是估计的唯一率,$U_{\mathrm{out1}} \in R^{H \times 0.5H}$,$b_{\mathrm{out1}} \in R^{0.5H}$ 和 $U_{\mathrm{out2}} \in R^{0.5H \times 1}$,$b_{\mathrm{out1}} \in R^1$ 是学习的的权重和偏差。

如上所述,使用经验性强且快速的 ReLU 激活函数来评估所有神经网络中隐含层单元,唯一率的值分布在[0,1]范围内。在预测唯一率 R_{out} 时,应用第二层中的 Sigmoid 激活函数来输出该范围内的浮点值。特别地,不对 R_{out} 做任何特征化,并且使用真实唯一率的值来训练模型,而不需要任何中间的特征化步骤。

值得注意的是,本模型对于 DISTINCT 类查询,预测唯一行的比率是基于以下目的:希望在不改变现有基数估计模型的情况下扩展模型以支持 DISTINCT 查询。例如,给定 SPARQL 查询 Q,任意有限基数估计模型(估计结果中包含重复行的模型) M,设基数 $C = M(Q)$,C 包含重复项,通过执行 $R_{\mathrm{out}} \cdot C$ 即可得到集合论基数(不包含重复项基数)。

2.4 训练数据

使用专门的查询生成器获得初始训练语料库。生成初始语料库分为两个步骤。第一步,生成两种类型的图模式。对于星型子图模式,随机选取一个起始节点,然后从该起始节点模拟一个随机步长 m 次,得到大小为 m 的星形图模式。类似地,对于链模式,从随机选择的节点开始游走,并在大小达到 n 后停止。其中 m 和 n 的大小由缩小采样的比例因子来决定。第二步,将图模式转换为示例查询。示例查询由三重模式,条件子句和查询结果的真实基数构成。首先,生成足够数量的子图模式后,将图模式中包含的所有主语、谓语、宾语按行转换成三重模式。在迭代转换过程中,对于每个三重模式中未绑定的变量,将其加入候选变量集。其次,生成查询的条件子句,条件子句由 (var,op,val) 组成,其中 var 从候选变量集中随机生成一个变量,op 则由操作符集{>,=,<}随机生成单个操作符,val 的值根据子图中对应的主语、谓语、宾语的值域随机生成。特别地,当某个三重模式的候选变量集的大小大于 1 时,为该三重模式生成多个条件子句(≤候选变量集的大小),多个条件子句之间用 && 连接。最后,发送由三重模式组成的 SPARQL 查询获取真实基数,如果真实基数为 0,则表示生成的查询不合法,将其丢弃。第三步,样本分类。执行完上述步骤后,将所有包含条件子句的示例查询加入语料库 1,用于训练 FILTER 查询的基数。其余查询加入到语料库 2,用于估计唯一率。语料库 2 的训练样本由三重模式,查询结果的真实唯一行比率组成。通过以上步骤,最终得到了文中模型的初始训练集。

3 实验与分析

3.1 实验设置

使用 LUBM[27] 数据集进行实验。为模拟 SPARQL 联邦查询,首先将 LUBM 的 1 700 万条数据按谓词数 12,12,11 划分为三个数据集(LUBM 谓词总数为 35),然后通过随机选取最终得到 15 个谓词,划分后每个数据集的三元组总数依次为 9 101 646,11 507 508,7 082 141。最后在划分得到的数据集上生成 30 万个具有 0 到 2 个连接的随机查询和 1 000 个物化样本作为训练数据,并且将训练数据分为 90% 的训练样本和 10% 的验证样本。使用训练数据来训练模型并得到对应的真实基数。

此外,为了验证模型的基础能力和扩展能力,在划分的数据集上合成了 4 种不同的工作负载:(1)FilterCrd_1,具有 4 500 个唯一查询,用于验证模型关于范围筛选的估计能力;(2)FilterCrd_2,具有 500 个唯一查询,旨在验证模型能否扩展到 2 个以上连接;(3)DistinctCrd_1,具有 4 500 个唯一查询,用于验证模型估计唯一率的基础能力;(4)DistinctCrd_2,具有 500 个唯一查询,旨在验证模型能否扩展到 2 个以上连接。表 1 显示了不同工作负载中连接数量的分布。以下将该文用于联邦复杂查询的基数估计模型(Cardinality Estimation of Federated Complex Queries,CEFCQ)记为 CEFCQ,同时实验将 CostFed[23] 和 CEFQR[24] 作为对比基线。

表 1 4 种工作负载的连接分布

工作负载	0	1	2	3	4	总计
FilterCrd_1	1 500	1 500	1 500	0	0	4 500
FilterCrd_2	100	100	100	100	100	500
DistinctCrd_1	1 500	1 500	1 500	0	0	4 500
DistinctCrd_2	100	100	100	100	100	500

3.2 估计质量

在两个查询工作负载 FilterCrd_1 和 DistinctCrd_1 上验证 CEFCQ 模型的基础估计能力,图 4 展示了实验结果,其中盒须图中方框边界位于第 25/75 百分位,水平“胡须”线标记为中位数位置。总体而言,CEFCQ 的两个方法都优于 CostFed 和 CEFQR,并且 CEFCQ 表

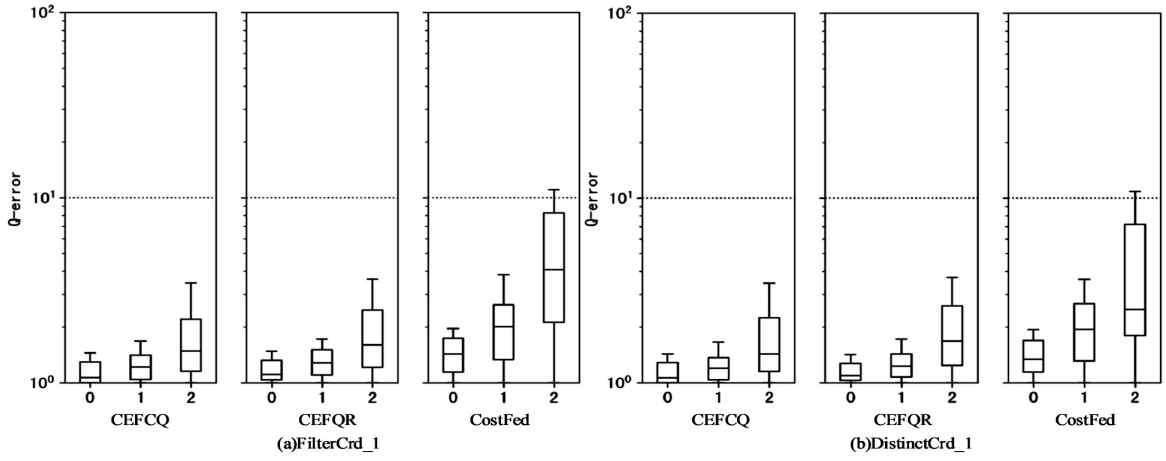

图 4 模型在不同工作负载上的估计误差盒须图

现得更稳健,同时具有更低的尾部误差。首先,相较于 CEFQR,CEFCQ 的提升虽然不是很明显,但其在扩展了查询类型的基础上仍能表现出优于 CEFQR 的估计质量,说明了 CEFCQ 的实用价值;其次,CEFQR 能提供较 CostFed 更精确的估计,这是由于 CEFQR 不依赖于从 SPARQL 端点收集的统计信息,使用监督学习模型能更准确地估计连接基数;另外,随着连接数量的增加,模型的估计质量在下降,这是因为估计多连接(连接数大于 0)查询时,基数值是累加的,可见 CEFCQ 对 0 个连接的查询估计质量最优。

为了提供更多详细信息,在表 2 和表 3 中分别显示了三个模型在以上工作负载上中位数、百分位数、最大 Q-error 和平均 Q-error。结果表明 CEFCQ 在各项指标上均表现优秀。

表 2　各模型在工作负载 FilterCrd_1 上的估计误差

Q-error	median	90th	95th	99th	max	mean
CEFCQ	1.17	2.8	3.2	6.23	91	1.62
CostFed	1.62	9	32	571.8	3 501	25.2
CEFQR	1.32	3.33	4.1	7	98.7	2.51

表 3　各模型在工作负载 DistinctCrd_1 上的估计误差

Q-error	median	90th	95th	99th	max	mean
CEFCQ	1.16	2.3	3.11	5.62	89.1	1.51
CostFed	1.5	9.2	27.98	569.2	3 210.88	23
CEFQR	1.2	2.9	3.41	6.11	93.2	1.72

3.3　扩展到更多连接

实验的目标是验证 CEFCQ 是否能够推广到连接数比训练时多的查询。因此,使用查询工作负载 FilterCrd_2 和 DistinctCrd_2(见表 1)来验证 CEFCQ 模型中两个方法的泛化能力。值得注意的是,在实验过程只使用具有 0 到 2 个连接的查询来训练 CEFCQ。图 5 的实验结果表明:整体来看,当连接数量大于训练时的最大连接数 2 时,CEFCQ 中两个方法的估计质量都有所下降,但是对比 CostFed,本模型仍然具有更好的可扩展性。

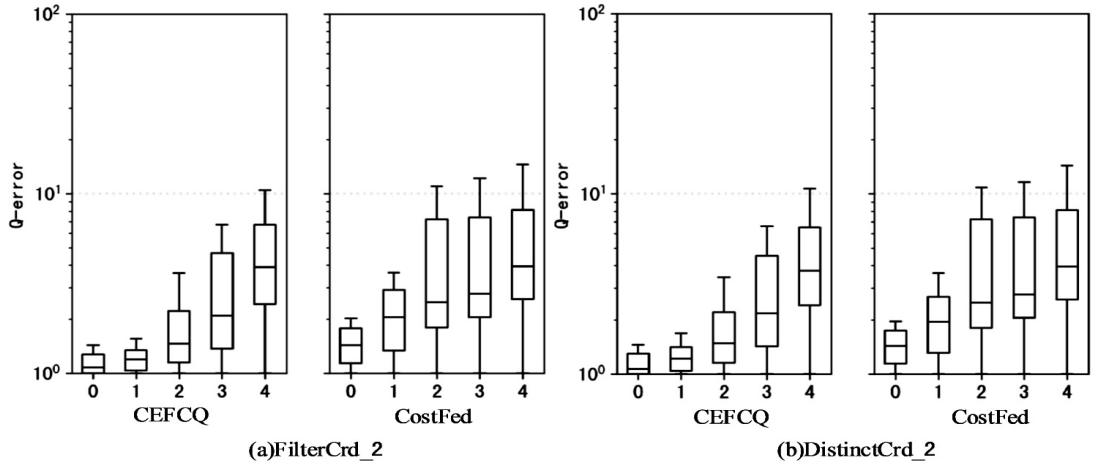

图 5　各模型在不同工作负载上的估计误差(展示 CEFCQ 如何泛化到具有更多连接的查询)

首先,在 FilterCrd_2 上验证模型中 FILTER 特征化方法的扩展性,当连接数大于 2 时,由于 CEFCQ 需要学习更多术语和图模式之间的相关性,从而降低了估计精度。连接数从 2 到 3 时,第 95 百分位的 Q-error 从 5.7 增加到 9.8,对比 CostFed 在相同查询上,第 95 百分位的 Q-error 为 94.3。当连接数为 4,第 95 个百分位数 Q-error 进一步增加到了 17.21(CostFed:560.3)。

其次,在 DistinctCrd_2 上验证 CEFCQ 学习唯一率的泛化能力,当连接数大于 2 时,CEFCQ 容易受到异常值的影响,虽然在训练模型时对数据进行了归一化和缩放,但偏度的影响仍然存在。连接数从 2 到 3 时,第 95 百分位的 Q-error 从 7.7 增加到 15.42(CostFed:133.2)。当连接数为 4,第 95 个百分位数 Q-error 增加到 28.34(CostFed:631.8)。作为参考,DistinctCrd_2 中的 500 个查询中有 48 个,超过了训练期间的最大唯一率。其中 32 个查询有 3 个连接,另有 16 个查询有 4 个连接。剔除这些异常值后,连接数为 3 和 4 的查询上,第 95 个百分位数的 Q-error 分别降至 10.2 和 24.5。

为了提供更多的细节,表 4 给出了 CEFCQ 和 CostFed 在两个工作负载上 Q-error 的中位数、最大值和平均值。可以看到 CEFCQ 在各个指标上均优于 CostFed。

表 4 各模型在工作负载 FilterCrd_2 和 DistinctCrd_2 上的估计误差

模型	FilterCrd_2			DistinctCrd_2		
	median	max	mean	median	max	mean
CEFCQ	1.8	40.2	4.21	1.72	38.4	3.13
CostFed	4.1	458 820.3	989.4	3.02	450 078.6	961.2

3.4 超参数和模型成本

为了优化 CEFCQ 的性能,搜索了超参数空间,考虑了不同的设置,其中改变了批次大小的数量(16,32,64,…,2 048)、隐藏层大小(16,32,64,…,1 024)和学习率(0.001,0.01)。检查了所有得到的 112 个不同的超参数组合。结果表明,隐藏层的大小对 CEFCQ 在验证测试中的准确性影响最大。在达到最佳结果之前,隐藏层的大小越大,CEFCQ 在验证测试中就越准确。之后,由于过度拟合,质量下降。此外,学习率和批次大小主要影响训练的收敛行为,而不是模型精度。在验证集上平均运行 5 次,最佳配置是:批次大小为 128、隐藏层大小 512 和 0.001 的学习率。因此,在文中模型评估中使用这些设置。在此设置下,CEFCQ 在训练集上运行大约 200 次后,在验证集上收敛到大约 3.7 的平均 Q-error(见图 6)。平均运行 5 次,200 个轮次(epochs)的训练阶段大约耗时 48 分钟。

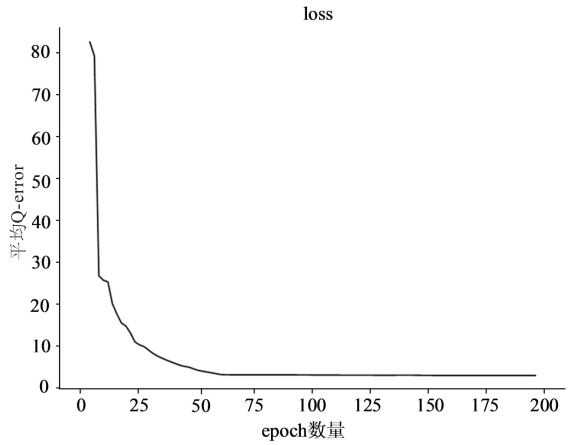

图 6 平均 Q-error 的收敛

4 结束语

该文提出了一种基于查询特征表示学习的联邦复杂查询基数估计模型方法。具体来说,主要考虑两类复杂查询,即 FILTER 条件筛选查询和 DISTINCT 查询。把估计这类查询基数的问题看做一个监督学习任务,提出一种在联邦查询中的监督学习模型。对于 FILTER 查询,提出一种 FILTER 特征化的技术,目的是将编码得到的特征向量作为模型的输入。在估计 DISTINCT 类查询时,模型输出估计的唯一率。对于这两种方法,进行了大量实验。实验结果表明,相较于之前的工作,该模型在基础估计能力和泛化能力上都得到了很大提升。结合现有工作的不足,未来研究将集中在两个方向。首先,扩展该模型以支持更多查询类型,例如带有 Top-k,GROUP BY,OPTIONAL 等操作符的查询。其次,由于该模型在估计结果时是基于 RDF 数据集是静态的假设下,但真实的 RDF 数据集是不定时更新的,因此当原始数据集发生变化时模型只能使用新的查询训练集重新训练,这会极大地增加计算成本。因此,下一步计划优化模型以支持数据集的增量更新。

参考文献:

[1] DAVITKOVA A, GJUROVSKI D, MICHEL S. LMKG: learned models for cardinality estimation in knowledge graphs[J]. arXiv:2102.10588,2021.

[2] KIPF A,FREITAG M,VORONA D,et al. Estimating filtered group-by queries is hard:deep learning to the rescue[C]//1st international workshop on applied AI for database systems and applications. Los Angeles:VLDB,2019.

[3] HAYEK R,SHMUELI O. Improved cardinality estimation by learning queries containment rates[J]. arXiv:1908.07723,2019.

[4] MÜLLER M,WOLTMANN L,LEHNER W. Enhanced featurization of queries with mixed combinations of predicates for ML-based cardinality estimation[C]//Proceedings of the 26th international conference on extending database technology. Ioannina:[s. n.],2022:273-284.

[5] BRESSAN M,PESERICO E,PRETTO L. Simple set cardinality estimation through random sampling[J]. arXiv:1512.07901,2015.

[6] LEIS V,RADKE B,GUBICHEV A,et al. Cardinality estimation done right:index-based join sampling[C]//Conference on innovative data systems research (CIDR). Chaminade:[s. n.],2017.

[7] KIPF A,KIPF T,RADKE B,et al. Learned cardinalities:estimating correlated joins with deep learning[J]. arXiv:1809.00677,2018.

[8] WOLTMANN L,HARTMANN C,THIELE M,et al. Cardinality estimation with local deep learning models[C]//Proceedings of the second international workshop on exploiting

artificial intelligence techniques for data management. New York:ACM,2019:1-8.

[9] DUTT A,WANG C,NAZI A,et al. Selectivity estimation for range predicates using lightweight models[J]. Proceedings of the VLDB Endowment,2019,12(9):1044-1057.

[10] HASAN S,THIRUMURUGANATHAN S,AUGUSTINE J,et al. Deep learning models for selectivity estimation of multi-attribute queries[C]//Proceedings of the 2020 ACM SIGMOD international conference on management of data. New York:ACM,2020:1035-1050.

[11] YANG Z,KAMSETTY A,LUAN S,et al. Neurocard:one cardinality estimator for all tables[J]. arXiv:2006.08109,2020.

[12] HASAN S,THIRUMURUGANATHAN S,AUGUSTINE J,et al. Multi-attribute selectivity estimation using deep learning[J]. arXiv:1903.09999,2019.

[13] LIU H,XU M,YU Z,et al. Cardinality estimation using neural networks[C]//Proceedings of the 25th annual international conference on computer science and software engineering. NJ:IBM Corp,2015:53-59.

[14] WANG X,QU C,WU W,et al. Are we ready for learned cardinality estimation? [J]. arXiv:2012.06743,2020.

[15] ZHU R,WU Z,HAN Y,et al. FLAT:fast,lightweight and accurate method for cardinality estimation[J]. arXiv:2011.09022,2020.

[16] WOLTMANN L,HARTMANN C,HABICH D,et al. Aggregate-based training phase for ML-based cardinality estimation[J]. Datenbank-Spektrum,2022,22(1):45-57.

[17] MARCUS R, PAPAEMMANOUIL O. Deep reinforcement learning for join order enumeration[C]//Proceedings of the first international workshop on exploiting artificial intelligence techniques for data management. New York:ACM,2018:1-4.

[18] YANG Z,LIANG E,KAMSETTY A,et al. Deep unsupervised cardinality estimation[J]. arXiv:1905.04278,2019.

[19] HILPRECHT B,SCHMIDT A,KULESSA M,et al. Deepdb:learn from data, not from queries! [J]. arXiv:1909.00607,2019.

[20] 乔少杰,杨国平,韩　楠,等. 基于树型门控循环单元的基数和代价估计器[J]. 软件学报,2022,33(3):797-813.

[21] GÖRLITZ O, STAAB S. SPLENDID:SPARQL endpoint federation exploiting VOID descriptions[C]//Proceedings of the second international conference on consuming linked data-volume 782. Aachen:CEUR-WS.org,2011:13-24.

[22] MONTOYA G,SKAF-MOLLI H,HOSE K. The Odyssey approach for optimizing federated SPARQL queries[C]// The semantic web - ISWC 2017. Vienna:Springer,2017:471-489.

[23] SALEEM M,POTOCKI A,SORU T,et al. CostFed:cost-based query optimization for SPARQL endpoint federation [J]. Procedia Computer Science,2018,137:163-174.

[24] 李　秋. 基于采样数据摘要和查询特征表示学习的联邦知识查询基数估计[D]. 武汉:武汉科技大学,2022.

[25] ABADI M,BARHAM P,CHEN J,et al. TensorFlow:a system for large-scale machine learning[J]. arXiv:1605.08695,2016.

[26] KINGMA D P,BA J. Adam:a method for stochastic optimization[J]. arXiv:1412.6980,2014.

[27] GUO Y,PAN Z,HEFLIN J. LUBM:a benchmark for OWL knowledge base systems[J]. Journal of Web Semantics,2005,3(2-3):158-182.