



(12) 发明专利

(10) 授权公告号 CN 112905591 B

(45) 授权公告日 2022. 08. 26

(21) 申请号 202110152193.2

张永清

(22) 申请日 2021.02.04

(74) 专利代理机构 北京正华智诚专利代理事务所(普通合伙) 11870

(65) 同一申请的已公布的文献号
申请公布号 CN 112905591 A

专利代理师 杨浩林

(43) 申请公布日 2021.06.04

(51) Int.Cl.

(73) 专利权人 成都信息工程大学
地址 610015 四川省成都市双流区西南航空港经济开发区学府路1段24号

G06F 16/22 (2019.01)

G06F 16/242 (2019.01)

G06N 3/08 (2006.01)

专利权人 成都探码科技有限公司
四川省金科成地理信息技术有限公司

(56) 对比文件

CN 108427985 A, 2018.08.21

CN 107239541 A, 2017.10.10

CN 109086267 A, 2018.12.25

CN 108959510 A, 2018.12.07

马鸿健等. 高校业务系统数据集成研究与应用.《信息通信》.2020, (第07期),

(72) 发明人 乔少杰 韩楠 宋学江 高瑞玮
肖月强 张小辉 赵兰 李鑫钰
冉先进 甘戈 孙科 范勇强
黄萍 魏军林 温敏 程维杰
叶青 余华 向导 彭京 周凯
元昌安 黄发良 覃晓 李斌勇

审查员 周循

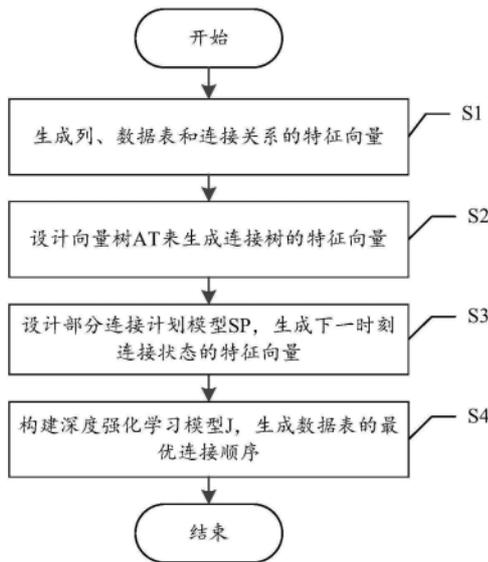
权利要求书6页 说明书16页 附图6页

(54) 发明名称

一种基于机器学习的数据表连接顺序选择方法

(57) 摘要

本发明公开了一种基于机器学习的数据表连接顺序选择方法,包括以下步骤:S1、对SQL语句进行编码,分别生成列、数据表和连接关系的特征向量;S2、根据列和数据表的特征向量,设计向量树AT来生成连接树的特征向量;S3、根据列、数据表、连接关系和连接树的特征向量,设计部分连接计划模型SP来生成部分连接计划的特征向量,进而生成下一时刻连接状态的特征向量;S4、根据下一时刻连接状态的特征向量,构建深度强化学习模型J,并结合部分连接计划模型SP与向量树AT,生成数据表的最优连接顺序。本发明解决了现有查询优化器生成数据表的次优连接顺序导致查询效率较低的问题。



1. 一种基于机器学习的数据表连接顺序选择方法,其特征在于,包括以下步骤:

S1、对SQL语句进行编码,分别生成列、数据表和连接关系的特征向量;

S2、根据列和数据表的特征向量,设计向量树AT来生成连接树的特征向量;

步骤S2包括以下分步骤:

S21、根据列的特征向量 $R(c)$ 和数据表的特征向量 $R(t)$,设计向量树AT将两个具有连接关系的数据表进行连接;

S22、将向量树AT的输入定义为左子节点和右子节点,将左子节点定义为 $(R(t_1), R(t_1.c))$,右子节点定义为 $(R(t_r), R(t_r.c))$,其中 t_1 和 t_r 为WHERE子句包含的数据表名,且数据表 t_1 和 t_r 具有连接关系, $t_1.c$ 和 $t_r.c$ 为WHERE子句包含的列名,且分别属于数据表 t_1 和 t_r , $R(t_1)$ 为数据表 t_1 的特征向量, $R(t_r)$ 为数据表 t_r 的特征向量, $R(t_1.c)$ 为列 $t_1.c$ 的特征向量, $R(t_r.c)$ 为列 $t_r.c$ 的特征向量;

S23、对向量树AT输入左子节点 $(R(t_1), R(t_1.c))$ 和右子节点 $(R(t_r), R(t_r.c))$,生成连接树的特征向量,计算公式为:

$$e = \text{sigmoid}(W_e * [R(t_1.c), R(t_r.c)] + Y_e * [R(t_1), R(t_r)] + b_e)$$

$$o = \text{tanh}(W_o * [R(t_1.c), R(t_r.c)] + Y_o * [R(t_1), R(t_r)] + b_o)$$

$$h = e * o$$

其中, $\text{sigmoid}()$ 和 $\text{tanh}()$ 为激活函数, e 为对左右子节点的信息融合更新的程度, o 为左右子节点被提取的信息, h 为连接树的特征向量, W_e 和 Y_e 为对信息融合更新的权重参数, b_e 为对信息融合更新的偏置参数, W_o 和 Y_o 为对信息进行提取的权重参数, b_o 为对信息进行提取的偏置参数;

S3、根据列、数据表、连接关系和连接树的特征向量,设计部分连接计划模型SP来生成部分连接计划的特征向量,进而生成下一时刻连接状态的特征向量;

步骤S3包括以下分步骤:

S31、根据列、数据表和连接树的特征向量,构建部分连接计划模型SP;

S32、根据部分连接计划模型SP,生成下一时刻部分连接计划的特征向量;

所述步骤S32具体为:

所述部分连接计划模型SP包括:第一类部分连接计划模型SP和第二类部分连接计划模型SP;

所述第一类部分连接计划模型SP将当前部分连接计划与单个数据表进行连接,且当前部分连接计划中某一个数据表与单个数据表具有连接关系;

将所述第一类部分连接计划模型SP的输入定义为第一类子节点和当前子节点,将第一类子节点定义为 $(R(t), R(t.c))$,其中, t 为WHERE子句包含的数据表名, $t.c$ 为WHERE子句包含的列名,且属于数据表 t , $R(t)$ 为数据表 t 的特征向量, $R(t.c)$ 为列 $t.c$ 的特征向量;将当前子节点定义为 $(H_i, R(t_H.c))$,其中, H 为部分连接计划的特征向量, H_i 为当前部分连接计划的特征向量, t_H 为当前部分连接计划包含的数据表名, $t_H.c$ 为数据表 t_H 包含的列名,且数据表 t 与数据表 t_H 具有连接关系;

对所述第一类部分连接计划模型SP输入第一类子节点和当前子节点,生成下一时刻部分连接计划的特征向量,计算公式为:

$$g_i = \text{sigmoid}(W_g * [R(t_H.c), R(t.c)] + Y_g * [H_i, R(t)] + b_g)$$

$$\begin{aligned} e_i &= \text{sigmoid}(W_e * [R(t_H.c), R(t.c)] + Y_e * [H_i, R(t)] + b_e) \\ o_i &= \text{tanh}(W_o * [R(t_H.c), R(t.c)] + Y_o * [g_i * H_i, R(t)] + b_o) \\ H_{i+1} &= (1 - e_i) * H_i + e_i * o_i \end{aligned}$$

其中, $\text{sigmoid}()$ 和 $\text{tanh}()$ 为激活函数, g_i 为当前子节点的信息没有被保留的程度, e_i 为对当前子节点和第一类子节点的信息提取的程度, o_i 为第一类子节点和当前子节点被保留的信息, H_{i+1} 为下一时刻部分连接计划的特征向量, W_g 和 Y_g 为对信息没有进行保留的权重参数, b_g 为对信息没有进行保留的偏置参数, W_e 和 Y_e 为对信息进行提取的权重参数, b_e 为对信息进行提取的偏置参数, W_o 和 Y_o 为对信息进行保留的权重参数, b_o 为对信息进行保留的偏置参数, W_e 、 W_o 、 Y_e 、 Y_o 、 b_e 、 b_o 与向量树AT共享;

所述第二类部分连接计划模型SP将当前部分连接计划与连接树进行连接, 且当前部分连接计划中某一个数据表与连接树中某一个数据表具有连接关系;

将所述第二类部分连接计划模型SP的输入定义为第二类子节点和当前子节点, 将第二类子节点定义为 $(h, R(t_h.c))$, 其中 h 为连接树的特征向量, t_h 为连接树包含的数据表名, $t_h.c$ 为数据表 t_h 包含的列名; 将当前子节点定义为 $(H_i, R(t_H.c))$, 其中, H 为部分连接计划的特征向量, H_i 为当前部分连接计划的特征向量, t_H 为当前部分连接计划包含的数据表名, $t_H.c$ 为数据表 t_H 包含的列名, 数据表 t_h 与数据表 t_H 具有连接关系;

对所述第二类部分连接计划模型SP输入第二类子节点和当前子节点, 生成下一时刻部分连接计划的特征向量, 计算公式为:

$$\begin{aligned} g_i &= \text{sigmoid}(W_g * [R(t_H.c), R(t_h.c)] + Y_g * [H_i, h] + b_g) \\ e_i &= \text{sigmoid}(W_e * [R(t_H.c), R(t_h.c)] + Y_e * [H_i, h] + b_e) \\ o_i &= \text{tanh}(W_o * [R(t_H.c), R(t_h.c)] + Y_o * [g_i * H_i, h] + b_o) \\ H_{i+1} &= (1 - e_i) * H_i + e_i * o_i \end{aligned}$$

其中, $\text{sigmoid}()$ 和 $\text{tanh}()$ 为激活函数, g_i 为对当前子节点的信息没有被保留的程度, e_i 为当前子节点与第二类子节点的信息提取的程度, o_i 为第二类子节点和当前子节点被保留的信息, H_{i+1} 为下一时刻部分连接计划的特征向量, W_g 和 Y_g 为对信息没有进行保留的权重参数, b_g 为对信息没有进行保留的偏置参数, W_e 和 Y_e 为对信息进行提取的权重参数, b_e 为对信息进行提取的偏置参数, W_o 和 Y_o 为对信息进行保留的权重参数, b_o 为对信息进行保留的偏置参数, W_e 、 W_o 、 Y_e 、 Y_o 、 b_e 、 b_o 与向量树AT共享;

S33、根据连接关系的特征向量和下一时刻部分连接计划的特征向量生成下一时刻连接状态的特征向量, 计算公式为:

$$R(s) = R(U) \oplus H_{i+1}$$

其中, $R(s)$ 为下一时刻连接状态 s 的特征向量, $R(U)$ 为连接关系的特征向量, H_{i+1} 为下一时刻部分连接计划的特征向量;

S4、根据下一时刻连接状态的特征向量, 构建深度强化学习模型J, 并结合部分连接计划模型SP与向量树AT, 生成数据表的最优连接顺序。

2. 根据权利要求1所述的基于机器学习的数据表连接顺序选择方法, 其特征在于, 所述步骤S1包括以下分步骤:

S11、对SQL语句中WHERE子句包含的列进行编码, 生成列的特征编码;

S12、对列的特征编码进行处理, 生成列的特征向量;

S13、根据列的特征向量,生成数据表的特征矩阵;

S14、根据数据表的特征矩阵,生成数据表的特征向量;

S15、对SQL语句中WHERE子句包含的连接关系进行编码,生成连接关系的特征编码;

S16、对连接关系的特征编码进行处理,生成连接关系的特征向量。

3. 根据权利要求2所述的基于机器学习的数据表连接顺序选择方法,其特征在于,所述步骤S11中对SQL语句中WHERE子句包含的列进行编码的信息包括:列关联的连接谓词和选择谓词;所述连接谓词包括:连接操作;所述选择谓词包括:等于、不等于、小于和大于的运算操作。

4. 根据权利要求3所述的基于机器学习的数据表连接顺序选择方法,其特征在于,所述步骤S11中连接谓词的编码公式为: $Join(c) = c_j$,其中,c为WHERE子句包含的任意一个列的列名,Join(c)为列关联的连接谓词的编码, c_j 为连接操作的编码, $c_j = 0$ 代表无连接操作, $c_j = 1$ 代表有连接操作;选择谓词的编码公式为: $Select(c) = (c_-, c_{!-}, c_<, c_>)$,其中,Select(c)为列关联的选择谓词的编码, c_- 为等于运算操作的编码, $c_{!-}$ 为不等于运算操作的编码, $c_<$ 为小于运算操作的编码, $c_>$ 为大于运算操作的编码,且 $c_-, c_{!-}, c_<, c_>$ 取值为0或者1,0代表无运算操作,1代表有运算操作;列的特征编码公式为: $F(c) = Select(c) \oplus Join(c)$,其中F(c)为列的特征编码, \oplus 为连接符;

所述步骤S12中生成列的特征向量的计算公式为:

$$R(c) = \text{leakyRelu}(F(c) * w + b)$$

其中,R(c)为列的特征向量,leakyRelu()为激活函数,F(c)为列的特征编码,w和b为参数;

所述步骤S13中生成数据表的特征矩阵的计算公式为:

$$T(t) = (R(c_1) \oplus \dots \oplus R(c_k))$$

其中,t为WHERE子句包含的任意一个数据表的表名,T(t)为数据表的特征矩阵, c_1, \dots, c_k 为WHERE子句包含的列名,且属于表t, $R(c_1), \dots, R(c_k)$ 为根据列的特征向量R(c)生成列 c_1, \dots, c_k 的特征向量;

步骤S14中生成数据表的特征向量的计算公式为:

$$R(t) = \text{MaxPool}(T(t))$$

其中,R(t)为数据表的特征向量,MaxPool()为最大池化函数;

步骤S15中生成连接关系的特征编码的公式如下:

$$u = (N_{1,2}, N_{1,3}, \dots, N_{1,n}, N_{2,3}, N_{2,4}, \dots, N_{2,n}, \dots, N_{a,b}, \dots, N_{n-1,n})$$

$a < b, 0 < a < n-1, 0 < b < n, a, b$ 均为整数

其中,u为连接关系的特征编码, $N_{1,2}, N_{1,3}, \dots, N_{1,n}, N_{2,3}, N_{2,4}, \dots, N_{2,n}, \dots, N_{a,b}, \dots, N_{n-1,n}$ 为WHERE子句包含的任意两个数据表的连接关系的编码, $N_{a,b}$ 为1时,代表第a个数据表与第b个数据表有连接, $N_{a,b}$ 为0时,代表第a个数据表与第b个数据表无连接,n为WHERE子句包含的数据表的数量;

所述步骤S16中生成连接关系的特征向量的计算公式为:

$$R(U) = \text{leakyRelu}(u * w + b)$$

其中,R(U)为连接关系的特征向量,leakyRelu()为激活函数,u为连接关系的特征编

码, w 和 b 为参数。

5. 根据权利要求1所述的基于机器学习的数据表连接顺序选择方法, 其特征在于, 所述步骤S4中深度强化学习模型J包括: 估计价值神经网络和目标价值神经网络;

所述估计价值神经网络和目标价值神经网络结构相同, 均包括: 输入层、卷积层、价值函数层、优势函数层和全连接层; 所述输入层的输入端作为估计价值神经网络或目标价值神经网络的输入端, 其输出端与卷积层的输入端连接; 所述卷积层的输出端分别与价值函数层的输入端和优势函数层的输入端连接; 所述价值函数层的输出端和优势函数层的输出端均与全连接层的输入端连接; 所述全连接层的输出端作为估计价值神经网络或目标价值神经网络的输出端。

6. 根据权利要求1所述的基于机器学习的数据表连接顺序选择方法, 其特征在于, 所述步骤S4包括以下分步骤:

S401、采用下一时刻部分连接计划的特征向量来重新代替新的当前部分连接计划的特征向量, 采用下一时刻连接状态的特征向量来重新代替新的当前连接状态的特征向量;

S402、将新的当前连接状态的特征向量输入估计价值神经网络, 得到当前连接状态下所有连接动作对应的估计价值;

S403、采用 ϵ -贪婪法对当前连接状态下所有连接动作对应的估计价值进行筛选, 得到当前连接状态下最小估计价值和当前连接状态下最小估计价值对应的连接动作, 其中, 连接动作表示在WHERE子句包含的数据表中选择两个具有连接关系的数据表进行连接;

S404、根据选择的两个具有连接关系的数据表和新的当前部分连接计划的特征向量, 结合向量树AT和部分连接计划模型SP, 来生成新的下一时刻部分连接计划的特征向量;

S405、根据步骤S404所得新的下一时刻部分连接计划的特征向量, 采用步骤S33的计算公式来生成新的下一时刻连接状态的特征向量;

S406、根据S405所得新的下一时刻连接状态的特征向量生成估计奖励;

S407、将步骤S401所得新的当前连接状态的特征向量、步骤S403所得最小估计价值对应的连接动作、步骤S405所得新的下一时刻连接状态的特征向量和步骤S406所得估计奖励组成单个样本放入训练集合ST中, 得到训练集合ST;

S408、根据步骤S405所得新的下一时刻连接状态的特征向量和步骤S406所得估计奖励, 计算目标估计价值;

S409、根据步骤S403所得最小估计价值和步骤S408所得目标估计价值, 从训练集合ST中采样 z 个样本, 构建均方差损失函数;

S410、根据均方差损失函数, 通过神经网络的梯度反向传播来更新估计价值神经网络的所有参数, 并且每隔额定时间, 将估计价值神经网络的所有参数的参数值传递给目标价值神经网络, 进而更新目标价值神经网络的所有参数;

S411、在更新目标价值神经网络所有参数的基础上, 根据步骤S404所得新的下一时刻部分连接计划的特征向量和步骤S405所得新的下一时刻连接状态的特征向量, 依次循环步骤S401、S402、S403、S404、S405、S406、S407、S408、S409和S410, 直到完整连接状态, 实现对深度强化学习模型J的所有参数完成一轮更新;

S412、采用多个SQL语句对深度强化学习模型J的所有参数完成多轮更新时, 得到深度强化学习模型J的最优参数, 生成数据表的最优连接顺序。

7. 根据权利要求6所述的基于机器学习的数据表连接顺序选择方法,其特征在于,所述步骤S404具体为:

当两个具有连接关系的数据表有且只有一个数据表不存在于当前部分连接计划中,采用第一类部分连接计划模型SP将这个数据表与当前部分连接计划进行连接,生成新的下一时刻部分连接计划的特征向量;当这两个数据表均不存在于当前部分连接计划中,但这两个数据表中某一个数据表与当前部分连接计划包含的某一个数据表具有连接关系时,采用向量树AT将所选择的两个具有连接关系的数据表进行连接,得到新的连接树,然后采用第二类部分连接计划模型SP将当前部分连接计划与新的连接树进行连接,生成新的下一时刻部分连接计划的特征向量。

8. 根据权利要求6所述的基于机器学习的数据表连接顺序选择方法,其特征在于,所述步骤S403中当前连接状态下最小估计价值和当前连接状态下最小估计价值对应的连接动作的计算公式为:

$$P_{est} = \min(P_{est}(R(s), A, w_{est}, \alpha, \beta))$$

$$A^{\min}(R(s), w_{est}, \alpha, \beta) = \operatorname{argmin} P_{est}(R(s), A, w_{est}, \alpha, \beta)$$

其中, P_{est} 为当前连接状态下最小估计价值, \min 为最小值函数, $P_{est}()$ 为估计价值神经网络, $R(s)$ 为新的当前连接状态 s 的特征向量, A 为连接动作, w_{est} 为估计价值神经网络的权重参数, α 为价值函数层的状态参数, β 为优势函数层的状态参数, $A^{\min}(R(s), w_{est}, \alpha, \beta)$ 为当前连接状态下最小估计价值对应的连接动作, argmin 为反函数;

所述步骤S406中生成估计奖励的计算公式为:

$$R = \operatorname{sigmod}(W_R * R(s)_{new})$$

其中, R 为估计奖励, $\operatorname{sigmod}()$ 为激活函数, $R(s)_{new}$ 为新的下一时刻连接状态 s 的特征向量, W_R 为计算估计奖励的参数;

所述步骤S408中计算目标估计价值需要使用下一时刻连接状态下最小估计价值对应的连接动作,计算公式为:

$$A^{\min}(R(s)_{new}, w_{est}, \alpha, \beta) = \operatorname{argmin} P_{est}(R(s)_{new}, A, w_{est}, \alpha, \beta)$$

其中, $A^{\min}(R(s)_{new}, w_{est}, \alpha, \beta)$ 为下一时刻连接状态下最小估计价值对应的连接动作, $R(s)_{new}$ 为新的下一时刻连接状态 s 的特征向量, w_{est} 为估计价值神经网络的权重参数, α 为价值函数层的状态参数, β 为优势函数层的状态参数, argmin 为反函数, $P_{est}()$ 为估计价值神经网络, A 为连接动作;

所述步骤S408中目标估计价值的计算公式为:

$$P_{target} = R + \gamma * P_{target}(R(s)_{new}, A^{\min}(R(s)_{new}, w_{est}, \alpha, \beta), w_{target})$$

其中, P_{target} 为目标估计价值, R 为步骤S405所得估计奖励, γ 为衰减程度因子, 取值范围为0到1, $P_{target}()$ 为目标价值神经网络, $R(s)_{new}$ 为新的下一时刻连接状态 s 的特征向量, $A^{\min}(R(s)_{new}, w_{est}, \alpha, \beta)$ 为下一时刻连接状态下最小估计价值对应的连接动作, w_{est} 为估计价值神经网络的权重参数, α 为价值函数层的状态参数, β 为优势函数层的状态参数, w_{target} 为目标价值神经网络的目标权重参数;

所述步骤S409构建均方差损失函数需要计算样本的采样优先级、样本的采样概率和样本的权重;

所述样本的采样优先级的计算公式为:

$$\&_r = |P_{\text{target}} - P_{\text{est}}|$$

其中, r 为样本序号, $\&_r$ 为第 r 个样本的采样优先级, $|\cdot|$ 为绝对值函数, P_{target} 为目标估计价值, P_{est} 为最小估计价值;

所述样本的采样概率的计算公式为:

$$P(r) = \&_r / (\&_1 + \dots + \&_z), r \leq z$$

其中, r 为样本序号, z 为样本数量, $P(r)$ 为第 r 个样本的采样概率, $\&_r$ 为第 r 个样本的采样优先级, $\&_1, \dots, \&_z$ 为第 1 个样本的采样优先级至第 z 个样本的采样优先级;

所述样本的权重的计算公式为:

$$w_r = (P(r))^{-z} / \max(w_1, \dots, w_z)$$

其中, r 为样本序号, z 为样本数量, w_r 为第 r 个样本的权重, $P(r)$ 为第 r 个样本的采样概率, $\max()$ 为最大值函数, w_1, \dots, w_z 为第 1 个样本的权重至第 z 个样本的权重;

所述步骤 S409 中均方差损失函数的计算公式为:

$$L = 1/z \sum_{r=1}^z w_r * (P_{\text{target}} - P_{\text{est}})^2$$

其中, r 为样本序号, z 为样本数量, L 为均方差损失函数值, Σ 为求和函数, w_r 为第 r 个样本的权重, P_{target} 为目标估计价值, P_{est} 为最小估计价值。

一种基于机器学习的数据表连接顺序选择方法

技术领域

[0001] 本发明涉及数据库领域,具体涉及一种基于机器学习的数据表连接顺序选择方法。

背景技术

[0002] 查询操作是关系数据库的基本操作,查询的效率是数据库系统的重要指标。于是查询优化成为数据库领域的重要研究方向。查询优化是为了构建一个代价最小的执行计划,从而达到在查询执行过程中的真实查询时间最低,实现用户的良好体验。通常,查询优化包括基数估计、代价模型、连接顺序选择等方面。查询优化的效果直接影响数据库应用程序的性能。

[0003] 而数据库查询优化器的重难点是多表连接顺序选择的问题,解决这个问题很复杂,代价巨大。因为它涉及到了为每个可选的连接顺序计算成本,从而极大地影响了查询优化器的优化过程。

[0004] 传统的数据库的优化方法基本上是基于基数估计和代价模型,采用动态规划搜索所有可能的连接顺序方案,虽然可以得到较好的方案,但搜索空间巨大,花费的代价令人望而却步。而采用剪枝技术加上启发式方法,虽然产生执行计划很快,但是通常会得到糟糕的计划。

发明内容

[0005] 针对现有技术中的上述不足,本发明提供了一种基于机器学习的数据表连接顺序选择方法解决了现有查询优化器生成数据表的次优连接顺序导致查询效率较低的问题。

[0006] 为了达到上述发明目的,本发明采用的技术方案为:一种基于机器学习的数据表连接顺序选择方法,包括以下步骤:

[0007] S1、对SQL语句进行编码,分别生成列、数据表和连接关系的特征向量;

[0008] S2、根据列和数据表的特征向量,设计向量树AT来生成连接树的特征向量;

[0009] S3、根据列、数据表、连接关系和连接树的特征向量,设计部分连接计划模型SP来生成部分连接计划的特征向量,进而生成下一时刻连接状态的特征向量;

[0010] S4、根据下一时刻连接状态的特征向量,构建深度强化学习模型J,并结合部分连接计划模型SP与向量树AT,生成数据表的最优连接顺序。

[0011] 进一步地,步骤S1包括以下分步骤:

[0012] S11、对SQL语句中WHERE子句包含的列进行编码,生成列的特征编码;

[0013] S12、对列的特征编码进行处理,生成列的特征向量;

[0014] S13、根据列的特征向量,生成数据表的特征矩阵;

[0015] S14、根据数据表的特征矩阵,生成数据表的特征向量;

[0016] S15、对SQL语句中WHERE子句包含的连接关系进行编码,生成连接关系的特征编码;

[0017] S16、对连接关系的特征编码进行处理,生成连接关系的特征向量。

[0018] 上述进一步方案的有益效果为:在数据预处理阶段,对SQL语句的信息采用特征向量进行表示,能够更适合后续机器学习模型的数据形式;采用特征向量不仅能够更准确地提取关键信息,而且能够包括许多SQL语句的信息,为后续步骤提供准确且全面的数据特征。

[0019] 进一步地,步骤S11中对SQL语句中WHERE子句包含的列进行编码的信息包括:列关联的连接谓词和选择谓词;所述连接谓词包括:连接操作;所述选择谓词包括:等于、不等于、小于和大于的运算操作。

[0020] 进一步地,步骤S11中连接谓词的编码公式为:Join(c) = c_j ,其中,c为WHERE子句包含的任意一个列的列名,Join(c)为列关联的连接谓词的编码, c_j 为连接操作的编码, $c_j=0$ 代表无连接操作, $c_j=1$ 代表有连接操作;选择谓词的编码公式为:Select(c) = ($c_=, c_{!}=, c_<, c_>$),其中,Select(c)为列关联的选择谓词的编码, $c_=$ 为等于运算操作的编码, $c_{!}=$ 为不等于运算操作的编码, $c_<$ 为小于运算操作的编码, $c_>$ 为大于运算操作的编码,且 $c_=, c_{!}=, c_<, c_>$ 取值为0或者1,0代表无运算操作,1代表有运算操作;列的特征编码公式为:
 $F(c)=Select(c)\oplus Join(c)$,其中F(c)为列的特征编码, \oplus 为连接符;

[0021] 所述步骤S12中生成列的特征向量的计算公式为:

[0022] $R(c) = \text{leakyRelu}(F(c) * w + b)$

[0023] 其中,R(c)为列的特征向量,leakyRelu()为激活函数,F(c)为列的特征编码,w和b为参数;

[0024] 所述步骤S13中生成数据表的特征矩阵的计算公式为:

[0025] $T(t) = (R(c_1) \oplus \dots \oplus R(c_k))$

[0026] 其中,t为WHERE子句包含的任意一个数据表的表名,T(t)为数据表的特征矩阵, c_1, \dots, c_k 为WHERE子句包含的列名,且属于表t, $R(c_1), \dots, R(c_k)$ 为根据列的特征向量R(c)生成列 c_1, \dots, c_k 的特征向量;

[0027] 步骤S14中生成数据表的特征向量的计算公式为:

[0028] $R(t) = \text{MaxPool}(T(t))$

[0029] 其中,R(t)为数据表的特征向量,MaxPool()为最大池化函数;

[0030] 步骤S15中生成连接关系的特征编码的公式如下:

[0031] $u = (N_{1,2}, N_{1,3}, \dots, N_{1,n}, N_{2,3}, N_{2,4}, \dots, N_{2,n}, \dots, N_{a,b}, \dots, N_{n-1,n})$

[0032] $a < b, 0 < a < n-1, 0 < b < n, a, b$ 均为整数

[0033] 其中,u为连接关系的特征编码, $N_{1,2}, N_{1,3}, \dots, N_{1,n}, N_{2,3}, N_{2,4}, \dots, N_{2,n}, \dots, N_{a,b}, \dots, N_{n-1,n}$ 为WHERE子句包含的任意两个数据表的连接关系的编码, $N_{a,b}$ 为1时,代表第a个数据表与第b个数据表有连接, $N_{a,b}$ 为0时,代表第a个数据表与第b个数据表无连接,n为WHERE子句包含的数据表的数量;

[0034] 所述步骤S16中生成连接关系的特征向量的计算公式为:

[0035] $R(U) = \text{leakyRelu}(u * w + b)$

[0036] 其中,R(U)为连接关系的特征向量,leakyRelu()为激活函数,u为连接关系的特征编码,w和b为参数。

[0037] 上述进一步方案的有益效果为：对特征编码进行处理能够避免特征编码的稀疏性，对特征矩阵进行处理能够最大限度地采样到关键信息来生成数据表的特征向量，进而能够匹配列的特征向量的维度，为后续步骤提供更加准确且全面的数据特征。

[0038] 进一步地，步骤S2包括以下分步骤：

[0039] S21、根据列的特征向量 $R(c)$ 和数据表的特征向量 $R(t)$ ，设计向量树AT将两个具有连接关系的数据表进行连接；

[0040] S22、将向量树AT的输入定义为左子节点和右子节点，将左子节点定义为 $(R(t_l), R(t_l.c))$ ，右子节点定义为 $(R(t_r), R(t_r.c))$ ，其中 t_l 和 t_r 为WHERE子句包含的数据表名，且数据表 t_l 和 t_r 具有连接关系， $t_l.c$ 和 $t_r.c$ 为WHERE子句包含的列名，且分别属于数据表 t_l 和 t_r ， $R(t_l)$ 为数据表 t_l 的特征向量， $R(t_r)$ 为数据表 t_r 的特征向量， $R(t_l.c)$ 为列 $t_l.c$ 的特征向量， $R(t_r.c)$ 为列 $t_r.c$ 的特征向量；

[0041] S23、对向量树AT输入左子节点 $(R(t_l), R(t_l.c))$ 和右子节点 $(R(t_r), R(t_r.c))$ ，生成连接树的特征向量，计算公式为：

$$[0042] \quad e = \text{sigmod}(W_e * [R(t_l.c), R(t_r.c)] + Y_e * [R(t_l), R(t_r)] + b_e)$$

$$[0043] \quad o = \text{tanh}(W_o * [R(t_l.c), R(t_r.c)] + Y_o * [R(t_l), R(t_r)] + b_o)$$

$$[0044] \quad h = e * o$$

[0045] 其中， $\text{sigmod}()$ 和 $\text{tanh}()$ 为激活函数， e 为对左右子节点的信息融合更新的程度， o 为左右子节点被提取的信息， h 为连接树的特征向量， W_e 和 Y_e 为对信息融合更新的权重参数， b_e 为对信息融合更新的偏置参数， W_o 和 Y_o 为对信息进行提取的权重参数， b_o 为对信息进行提取的偏置参数。

[0046] 上述进一步方案的有益效果为：采用向量树AT生成连接树的特征向量，能够更加符合两个具有连接关系的数据表进行连接后的结果，为后续构建部分连接计划提供基本元素。

[0047] 进一步地，步骤S3包括以下分步骤：

[0048] S31、根据列、数据表和连接树的特征向量，构建部分连接计划模型SP；

[0049] S32、根据部分连接计划模型SP，生成下一时刻部分连接计划的特征向量；

[0050] 所述步骤S32具体为：

[0051] 所述部分连接计划模型SP包括：第一类部分连接计划模型SP和第二类部分连接计划模型SP；

[0052] 所述第一类部分连接计划模型SP将当前部分连接计划与单个数据表进行连接，且当前部分连接计划中某一个数据表与单个数据表具有连接关系；

[0053] 将所述第一类部分连接计划模型SP的输入定义为第一类子节点和当前子节点，将第一类子节点定义为 $(R(t), R(t.c))$ ，其中， t 为WHERE子句包含的数据表名， $t.c$ 为WHERE子句包含的列名，且属于数据表 t ， $R(t)$ 为数据表 t 的特征向量， $R(t.c)$ 为列 $t.c$ 的特征向量；将当前子节点定义为 $(H_i, R(t_H.c))$ ，其中， H 为部分连接计划的特征向量， H_i 为当前部分连接计划的特征向量， t_H 为当前部分连接计划包含的数据表名， $t_H.c$ 为数据表 t_H 包含的列名，且数据表 t 与数据表 t_H 具有连接关系；

[0054] 对所述第一类部分连接计划模型SP输入第一类子节点和当前子节点，生成下一时刻部分连接计划的特征向量，计算公式为：

$$[0055] \quad g_i = \text{sigmoid}(W_g * [R(t_H.c), R(t.c)] + Y_g * [H_i, R(t)] + b_g)$$

$$[0056] \quad e_i = \text{sigmoid}(W_e * [R(t_H.c), R(t.c)] + Y_e * [H_i, R(t)] + b_e)$$

$$[0057] \quad o_i = \text{tanh}(W_o * [R(t_H.c), R(t.c)] + Y_o * [g_i * H_i, R(t)] + b_o)$$

$$[0058] \quad H_{i+1} = (1 - e_i) * H_i + e_i * o_i$$

[0059] 其中, $\text{sigmoid}()$ 和 $\text{tanh}()$ 为激活函数, g_i 为当前子节点的信息没有被保留的程度, e_i 为对当前子节点和第一类子节点的信息提取的程度, o_i 为第一类子节点和当前子节点被保留的信息, H_{i+1} 为下一时刻部分连接计划的特征向量, W_g 和 Y_g 为对信息没有进行保留的权重参数, b_g 为对信息没有进行保留的偏置参数, W_e 和 Y_e 为对信息进行提取的权重参数, b_e 为对信息进行提取的偏置参数, W_o 和 Y_o 为对信息进行保留的权重参数, b_o 为对信息进行保留的偏置参数, W_e 、 W_o 、 Y_e 、 Y_o 、 b_e 、 b_o 与向量树AT共享;

[0060] 所述第二类部分连接计划模型SP将当前部分连接计划与连接树进行连接, 且当前部分连接计划中某一个数据表与连接树中某一个数据表具有连接关系;

[0061] 将所述第二类部分连接计划模型SP的输入定义为第二类子节点和当前子节点, 将第二类子节点定义为 $(h, R(t_h.c))$, 其中 h 为连接树的特征向量, t_h 为连接树包含的数据表名, $t_h.c$ 为数据表 t_h 包含的列名; 将当前子节点定义为 $(H_i, R(t_H.c))$, 其中, H 为部分连接计划的特征向量, H_i 为当前部分连接计划的特征向量, t_H 为当前部分连接计划包含的数据表名, $t_H.c$ 为数据表 t_H 包含的列名, 数据表 t_h 与数据表 t_H 具有连接关系;

[0062] 对所述第二类部分连接计划模型SP输入第二类子节点和当前子节点, 生成下一时刻部分连接计划的特征向量, 计算公式为:

$$[0063] \quad g_i = \text{sigmoid}(W_g * [R(t_H.c), R(t_h.c)] + Y_g * [H_i, h] + b_g)$$

$$[0064] \quad e_i = \text{sigmoid}(W_e * [R(t_H.c), R(t_h.c)] + Y_e * [H_i, h] + b_e)$$

$$[0065] \quad o_i = \text{tanh}(W_o * [R(t_H.c), R(t_h.c)] + Y_o * [g_i * H_i, h] + b_o)$$

$$[0066] \quad H_{i+1} = (1 - e_i) * H_i + e_i * o_i$$

[0067] 其中, $\text{sigmoid}()$ 和 $\text{tanh}()$ 为激活函数, g_i 为对当前子节点的信息没有被保留的程度, e_i 为当前子节点与第二类子节点的信息提取的程度, o_i 为第二类子节点和当前子节点被保留的信息, H_{i+1} 为下一时刻部分连接计划的特征向量, W_g 和 Y_g 为对信息没有进行保留的权重参数, b_g 为对信息没有进行保留的偏置参数, W_e 和 Y_e 为对信息进行提取的权重参数, b_e 为对信息进行提取的偏置参数, W_o 和 Y_o 为对信息进行保留的权重参数, b_o 为对信息进行保留的偏置参数, W_e 、 W_o 、 Y_e 、 Y_o 、 b_e 、 b_o 与向量树AT共享;

[0068] S33、根据连接关系的特征向量和下一时刻部分连接计划的特征向量生成下一时刻连接状态的特征向量, 计算公式为:

$$[0069] \quad R(s) = R(U) \oplus H_{i+1}$$

[0070] 其中, $R(s)$ 为下一时刻连接状态 s 的特征向量, $R(U)$ 为连接关系的特征向量, H_{i+1} 为下一时刻部分连接计划的特征向量。

[0071] 上述进一步方案的有益效果为: 采用部分连接计划模型SP生成下一时刻部分连接计划的特征向量, 能够将部分连接计划的结构信息包含进去。结合后续步骤能够不断更新下一时刻部分连接计划, 使部分连接计划的结构能够进行动态扩展, 从而更加精准地捕捉到部分连接计划的结构信息, 能够更加准确生成下一时刻连接状态的特征向量。

[0072] 进一步地, 步骤S4中深度强化学习模型J包括: 估计价值神经网络和目标价值神经

网络；

[0073] 所述估计价值神经网络和目标价值神经网络结构相同，均包括：输入层、卷积层、价值函数层、优势函数层和全连接层；所述输入层的输入端作为估计价值神经网络或目标价值神经网络的输入端，其输出端与卷积层的输入端连接；所述卷积层的输出端分别与价值函数层的输入端和优势函数层的输入端连接；所述价值函数层的输出端和优势函数层的输出端均与全连接层的输入端连接；所述全连接层的输出端作为估计价值神经网络或目标价值神经网络的输出端。

[0074] 进一步地，步骤S4包括以下分步骤：

[0075] S401、采用下一时刻部分连接计划的特征向量来重新代替新的当前部分连接计划的特征向量，采用下一时刻连接状态的特征向量来重新代替新的当前连接状态的特征向量；

[0076] S402、将新的当前连接状态的特征向量输入估计价值神经网络，得到当前连接状态下所有连接动作对应的估计价值；

[0077] S403、采用 ϵ -贪婪法对当前连接状态下所有连接动作对应的估计价值进行筛选，得到当前连接状态下最小估计价值和当前连接状态下最小估计价值对应的连接动作，其中，连接动作表示在WHERE子句包含的数据表中选择两个具有连接关系的数据表进行连接；

[0078] S404、根据选择的两个具有连接关系的数据表和新的当前部分连接计划的特征向量，结合向量树AT和部分连接计划模型SP，来生成新的下一时刻部分连接计划的特征向量；

[0079] S405、根据步骤S404所得新的下一时刻部分连接计划的特征向量，采用步骤S33的计算公式来生成新的下一时刻连接状态的特征向量；

[0080] S406、根据S405所得新的下一时刻连接状态的特征向量生成估计奖励；

[0081] S407、将步骤S401所得新的当前连接状态的特征向量、步骤S403所得最小估计价值对应的连接动作、步骤S405所得新的下一时刻连接状态的特征向量和步骤S406所得估计奖励组成单个样本放入训练集合ST中，得到训练集合ST；

[0082] S408、根据步骤S405所得新的下一时刻连接状态的特征向量和步骤S406所得估计奖励，计算目标估计价值；

[0083] S409、根据步骤S403所得最小估计价值和步骤S408所得目标估计价值，从训练集合ST中采样z个样本，构建均方差损失函数；

[0084] S410、根据均方差损失函数，通过神经网络的梯度反向传播来更新估计价值神经网络的所有参数，并且每隔额定时间，将估计价值神经网络的所有参数的参数值传递给目标价值神经网络，进而更新目标价值神经网络的所有参数；

[0085] S411、在更新目标价值神经网络所有参数的基础上，根据步骤S404所得新的下一时刻部分连接计划的特征向量和步骤S405所得新的下一时刻连接状态的特征向量，依次循环步骤S401、S402、S403、S404、S405、S406、S407、S408、S409和S410，直到完整连接状态，实现对深度强化学习模型J的所有参数完成一轮更新；

[0086] S412、采用多个SQL语句对深度强化学习模型J的所有参数完成多轮更新时，得到深度强化学习模型J的最优参数，生成数据表的最优连接顺序。

[0087] 上述进一步方案的有益效果为：估计价值神经网络能够直接得到连接动作，相比动态规划方法能够极大降低搜索代价。而且神经网络经过训练后，能够学习到糟糕的连接

计划,就能避免之后出现同样糟糕的连接计划,这是传统查询优化器所不具备的学习经验。更为关键的是,整个深度强化学习模型J经过训练后能够超出传统查询优化器关于连接顺序选择的性能,解决查询时间缓慢的问题。

[0088] 进一步地,步骤S404具体为:

[0089] 当两个具有连接关系的数据表有且只有一个数据表不存在于当前部分连接计划中,采用第一类部分连接计划模型SP将这个数据表与当前部分连接计划进行连接,生成新的下一时刻部分连接计划的特征向量;当这两个数据表均不存在于当前部分连接计划中,但这两个数据表中某一个数据表与当前部分连接计划包含的某一个数据表具有连接关系时,采用向量树AT将所选择的两个具有连接关系的数据表进行连接,得到新的连接树,然后采用第二类部分连接计划模型SP将当前部分连接计划与新的连接树进行连接,生成新的下一时刻部分连接计划的特征向量。

[0090] 进一步地,步骤S403中当前连接状态下最小估计价值和当前连接状态下最小估计价值对应的连接动作的计算公式为:

$$[0091] \quad P_{est} = \min(P_{est}(R(s), A, w_{est}, \alpha, \beta))$$

$$[0092] \quad A^{\min}(R(s), w_{est}, \alpha, \beta) = \operatorname{argmin} P_{est}(R(s), A, w_{est}, \alpha, \beta)$$

[0093] 其中, P_{est} 为当前连接状态下最小估计价值, \min 为最小值函数, $P_{est}()$ 为估计价值神经网络, $R(s)$ 为新的当前连接状态s的特征向量, A 为连接动作, w_{est} 为估计价值神经网络的权重参数, α 为价值函数层的状态参数, β 为优势函数层的状态参数, $A^{\min}(R(s), w_{est}, \alpha, \beta)$ 为当前连接状态下最小估计价值对应的连接动作, argmin 为反函数;

[0094] 所述步骤S406中生成估计奖励的计算公式为:

$$[0095] \quad R = \operatorname{sigmod}(W_R * R(s)_{new})$$

[0096] 其中, R 为估计奖励, $\operatorname{sigmod}()$ 为激活函数, $R(s)_{new}$ 为新的下一时刻连接状态s的特征向量, W_R 为计算估计奖励的参数;

[0097] 所述步骤S408中计算目标估计价值需要使用下一时刻连接状态下最小估计价值对应的连接动作,计算公式为:

$$[0098] \quad A^{\min}(R(s)_{new}, w_{est}, \alpha, \beta) = \operatorname{argmin} P_{est}(R(s)_{new}, A, w_{est}, \alpha, \beta)$$

[0099] 其中, $A^{\min}(R(s)_{new}, w_{est}, \alpha, \beta)$ 为下一时刻连接状态下最小估计价值对应的连接动作, $R(s)_{new}$ 为新的下一时刻连接状态s的特征向量, w_{est} 为估计价值神经网络的权重参数, α 为价值函数层的状态参数, β 为优势函数层的状态参数, argmin 为反函数, $P_{est}()$ 为估计价值神经网络, A 为连接动作;

[0100] 所述步骤S408中目标估计价值的计算公式为:

$$[0101] \quad P_{target} = R + \gamma * P_{target}(R(s)_{new}, A^{\min}(R(s)_{new}, w_{est}, \alpha, \beta), w_{target})$$

[0102] 其中, P_{target} 为目标估计价值, R 为步骤S405所得估计奖励, γ 为衰减程度因子,取值范围为0到1, $P_{target}()$ 为目标价值神经网络, $R(s)_{new}$ 为新的下一时刻连接状态s的特征向量, $A^{\min}(R(s)_{new}, w_{est}, \alpha, \beta)$ 为下一时刻连接状态下最小估计价值对应的连接动作, w_{est} 为估计价值神经网络的权重参数, α 为价值函数层的状态参数, β 为优势函数层的状态参数, w_{target} 为目标价值神经网络的目标权重参数;

[0103] 所述步骤S409构建均方差损失函数需要计算样本的采样优先级、样本的采样概率和样本的权重;

[0104] 所述样本的采样优先级的计算公式为：

$$[0105] \quad \&_r = |P_{\text{target}} - P_{\text{est}}|$$

[0106] 其中， r 为样本序号， $\&_r$ 为第 r 个样本的采样优先级， $|\cdot|$ 为绝对值函数， P_{target} 为目标估计价值， P_{est} 为最小估计价值；

[0107] 所述样本的采样概率的计算公式为：

$$[0108] \quad P(r) = \&_r / (\&_1 + \dots + \&_z), r \leq z$$

[0109] 其中， r 为样本序号， z 为样本数量， $P(r)$ 为第 r 个样本的采样概率， $\&_r$ 为第 r 个样本的采样优先级， $\&_1, \dots, \&_z$ 为第1个样本的采样优先级至第 z 个样本的采样优先级；

[0110] 所述样本的权重的计算公式为：

$$[0111] \quad w_r = (P(r))^{-z} / \max(w_1, \dots, w_z)$$

[0112] 其中， r 为样本序号， z 为样本数量， w_r 为第 r 个样本的权重， $P(r)$ 为第 r 个样本的采样概率， $\max()$ 为最大值函数， w_1, \dots, w_z 为第1个样本的权重至第 z 个样本的权重；

[0113] 所述步骤S409中均方差损失函数的计算公式为：

$$[0114] \quad L = 1/z \sum_{r=1}^z w_r * (P_{\text{target}} - P_{\text{est}})^2$$

[0115] 其中， r 为样本序号， z 为样本数量， L 为均方差损失函数值， Σ 为求和函数， w_r 为第 r 个样本的权重， P_{target} 为目标估计价值， P_{est} 为最小估计价值。

[0116] 上述进一步方案的有益效果为：采用估计价值神经网络来得到连接动作，再采用目标价值神经网络来得到目标估计价值，相比采用一个神经网络而言，将得到连接动作的过程和计算目标估计价值的过程分开，能够优化偏低估计的现象。而且在两个神经网络中均采用优势函数层和价值函数层，可以优化神经网络的结构来尽可能使神经网络达到更加良好的效果。另外，在均方差损失函数中引入了样本的采样优先级，样本的采样概率，能够使具有较大误差的样本尽可能被采样到，从而加快了训练过程，能够使神经网络尽快收敛。

[0117] 综上，本发明的有益效果为：首先，在SQL语句的编码形式上更加全面，提供了更加准确的数据特征；其次能够捕捉到动态扩展的连接计划的结构信息，为深度强化学习提供了更加全面的连接状态信息；最后，在深度强化学习模型J中不仅设计了两个神经网络来优化偏低估计的现象，达到更好的效果，而且设计了价值函数层和优势函数层来优化神经网络结构，以及设计了样本的采样优先级和样本的采样概率来使训练加速，尽快达到收敛。本发明得到数据表的最优连接顺序，在真实查询过程中耗费时间更低，获得更好的查询性能。

附图说明

[0118] 图1为本方法的大致流程图；

[0119] 图2为本方法的详细流程图；

[0120] 图3为本发明的向量树AT架构图；

[0121] 图4为本发明的第一类部分连接计划模型SP的连接过程图；

[0122] 图5为本发明的第二类部分连接计划模型SP的连接过程图；

[0123] 图6为本发明的深度强化学习模型J架构图；

[0124] 图7为本发明实施例中的SQL语句例子；

[0125] 图8为本发明实施例中的新的当前部分连接计划1；

[0126] 图9为本发明实施例中的新的当前部分连接计划2；

[0127] 图10为本发明实施例中的数据表的最优连接顺序。

具体实施方式

[0128] 下面对本发明的具体实施方式进行了描述,以便于本技术领域的技术人员理解本发明,但应该清楚,本发明不限于具体实施方式的范围,对本技术领域的普通技术人员来讲,只要各种变化在所附的权利要求限定和确定的本发明的精神和范围内,这些变化是显而易见的,一切利用本发明构思的发明创造均在保护之列。

[0129] 如图1~2所示,一种基于机器学习的数据表连接顺序选择方法,包括以下步骤:

[0130] S1、对SQL语句进行编码,分别生成列、数据表和连接关系的特征向量;

[0131] 在本实施例中,将基于机器学习的数据表连接顺序选择方法在主流数据库上构建,替代传统连接顺序选择方法(如启发式、动态规划等)的查询优化组件,同时基于TPC-H建立包含8个表的销售供货数据库,数据量达到GB级,且通过包含22条SQL语句的模板生成多条SQL语句进行训练和测试,图7仅显示了其中包含一条SQL语句的例子。基于图7例子,对本发明的整体流程说明。首先,对图7中SQL语句中WHERE子句包含的列进行编码,即步骤S1包括以下分步骤:

[0132] S11、对SQL语句中WHERE子句包含的列进行编码,生成列的特征编码;

[0133] 步骤S11中对SQL语句中WHERE子句包含的列进行编码的信息包括:列关联的连接谓词和选择谓词;所述连接谓词包括:连接操作;所述选择谓词包括:等于、不等于、小于和大于的运算操作。

[0134] 步骤S11中连接谓词的编码公式为: $Join(c) = c_j$,其中, c 为WHERE子句包含的任意一个列的列名, $Join(c)$ 为列关联的连接谓词的编码, c_j 为连接操作的编码, $c_j=0$ 代表无连接操作, $c_j=1$ 代表有连接操作;选择谓词的编码公式为: $Select(c) = (c_-, c_{!-}, c_<, c_>)$,其中, $Select(c)$ 为列关联的选择谓词的编码, c_- 为等于运算操作的编码, $c_{!-}$ 为不等于运算操作的编码, $c_<$ 为小于运算操作的编码, $c_>$ 为大于运算操作的编码,且 $c_-, c_{!-}, c_<, c_>$ 取值为0或者1,0代表无运算操作,1代表有运算操作;列的特征编码公式为: $F(c) = Select(c) \oplus Join(c)$,其中 $F(c)$ 为列的特征编码, \oplus 为连接符;

[0135] 列的特征编码如表1所示。

[0136] 表1列的特征编码的表

列名	连接谓词的编码 $Join(c)$	选择谓词的编码 $Select(c)$
l_orderkey	1	0,0,0,0
o_orderkey	1	0,0,0,0
l_suppkey	1	0,0,0,0
s_suppkey	1	0,0,0,0

[0137]

[0138]	c_nationkey	1	0,0,0,0
	s_nationkey	1	0,0,0,0
	n_nationkey	1	0,0,0,0
	n_regionkey	1	0,0,0,0
	r_regionkey	1	0,0,0,0
	r_name	0	1,0,0,0
	o_orderdate	0	1,0,1,1

[0139] S12、对列的特征编码进行处理,生成列的特征向量;

[0140] 所述步骤S12中生成列的特征向量的计算公式为:

[0141] $R(c) = \text{leakyRelu}(F(c) * w + b)$

[0142] 其中, $R(c)$ 为列的特征向量, $\text{leakyRelu}()$ 为激活函数, $F(c)$ 为列的特征编码, w 和 b 为参数;

[0143] 在本实施例中,步骤S12具体为:基于步骤S11所得列的特征编码表1,对列的特征编码进行处理,消除特征编码的稀疏性,生成更加精确的特征向量,这里将数据表Lineitem包含的列orderkey、列suppkey和数据表Orders包含的列orderkey、列orderdate作为例子进行说明,公式如下:

[0144] $R(L.or) = \text{LeakyRelu}(F(L.or) * w + b)$

[0145] $R(L.su) = \text{LeakyRelu}(F(L.su) * w + b)$

[0146] $R(O.or) = \text{LeakyRelu}(F(O.or) * w + b)$

[0147] $R(O.od) = \text{LeakyRelu}(F(O.od) * w + b)$

[0148] 其中, $R(L.or)$ 为数据表Lineitem包含的列orderkey的特征向量, $F(L.or)$ 为数据表Lineitem包含的列orderkey的特征编码, $R(L.su)$ 为数据表Lineitem包含的列suppkey的特征向量, $F(L.su)$ 为数据表Lineitem包含的列suppkey的特征编码, $R(O.or)$ 为数据表Orders包含的列orderkey的特征向量, $F(O.or)$ 为数据表Orders包含的列orderkey的特征编码, $R(O.od)$ 为数据表Orders包含的列orderdate的特征向量, $F(O.od)$ 为数据表Orders包含的列orderdate的特征编码;

[0149] S13、根据列的特征向量,生成数据表的特征矩阵;

[0150] 步骤S13中生成数据表的特征矩阵的计算公式为:

[0151] $T(t) = (R(c_1) \oplus \dots \oplus R(c_k))$

[0152] 其中, t 为WHERE子句包含的任意一个数据表的表名, $T(t)$ 为数据表的特征矩阵, c_1, \dots, c_k 为WHERE子句包含的列名,且属于表 t , $R(c_1), \dots, R(c_k)$ 为根据列的特征向量 $R(c)$ 生成列 c_1, \dots, c_k 的特征向量;

[0153] 在本实施例中,步骤S13具体为:基于步骤S12所得数据表Lineitem包含的列orderkey的特征向量 $R(L.or)$ 、数据表Lineitem包含的列suppkey的特征向量 $R(L.su)$ 、数据表Orders包含的列orderkey的特征向量 $R(O.or)$ 、数据表Orders包含的列orderdate的特征向量 $R(O.od)$,生成数据表Lineitem的特征矩阵 $T(L)$ 和数据表Orders的特征矩阵 $T(O)$,公式如下:

[0154] $T(L)=R(L.or)\oplus R(L.su)$

[0155] $T(O)=R(O.or)\oplus R(O.od)$

[0156] S14、根据数据表的特征矩阵,生成数据表的特征向量;

[0157] 步骤S14中生成数据表的特征向量的计算公式为:

[0158] $R(t) = \text{MaxPool}(T(t))$

[0159] 其中, $R(t)$ 为数据表的特征向量, $\text{MaxPool}()$ 为最大池化函数;

[0160] 在本实施例中,步骤S14具体为:根据数据表Lineitem的特征矩阵 $T(L)$ 和数据表Orders的特征矩阵 $T(O)$,采用最大池化函数,生成数据表Lineitem的特征向量 $R(L)$ 和数据表Orders的特征向量 $R(O)$,公式如下:

[0161] $R(L) = \text{MaxPool}(T(L))$

[0162] $R(O) = \text{MaxPool}(T(O))$

[0163] S15、对SQL语句中WHERE子句包含的连接关系进行编码,生成连接关系的特征编码;

[0164] 步骤S15中生成连接关系的特征编码的公式如下:

[0165] $u = (N_{1,2}, N_{1,3}, \dots, N_{1,n}, N_{2,3}, N_{2,4}, \dots, N_{2,n}, \dots, N_{a,b}, \dots, N_{n-1,n})$

[0166] $a < b, 0 < a < n-1, 0 < b < n, a, b$ 均为整数

[0167] 其中, u 为连接关系的特征编码, $N_{1,2}, N_{1,3}, \dots, N_{1,n}, N_{2,3}, N_{2,4}, \dots, N_{2,n}, \dots, N_{a,b}, \dots, N_{n-1,n}$ 为WHERE子句包含的任意两个数据表的连接关系的编码, $N_{a,b}$ 为1时,代表第 a 个数据表与第 b 个数据表有连接, $N_{a,b}$ 为0时,代表第 a 个数据表与第 b 个数据表无连接, n 为WHERE子句包含的数据表的数量;

[0168] 在本实施例中,步骤S15具体为:将图7中SQL语句中WHERE子句包含的数据表Customer、Orders、Lineitem、Supplier、Nation、Region依次编号为1,2,3,4,5,6,易知数据表Customer与数据表Lineitem、数据表Supplier均有连接,数据表Lineitem与数据表Supplier有连接,数据表Supplier与数据表Nation有连接,数据表Nation与数据表Region有连接,可得 $u = (0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 1, 0, 1)$;

[0169] S16、对连接关系的特征编码进行处理,生成连接关系的特征向量;

[0170] 步骤S16中生成连接关系的特征向量的计算公式为:

[0171] $R(U) = \text{leakyRelu}(u*w+b)$

[0172] 其中, $R(U)$ 为连接关系的特征向量, $\text{leakyRelu}()$ 为激活函数, u 为连接关系的特征编码, w 和 b 为参数。

[0173] S2、根据列和数据表的特征向量,设计向量树AT来生成连接树的特征向量;

[0174] 在本实施例中,采用图7中SQL语句中WHERE子句包含的数据表Lineitem和数据表Orders作为例子进行说明,设计向量树AT将数据表Lineitem与数据表Orders进行连接,生成包含数据表Lineitem和数据表Orders的连接树的特征向量 h_{L0} ,向量树AT架构如图3所示。

[0175] 所述步骤S2包括以下分步骤:

[0176] S21、根据列的特征向量 $R(c)$ 和数据表的特征向量 $R(t)$,设计向量树AT将两个具有连接关系的数据表进行连接;

[0177] S22、将向量树AT的输入定义为左子节点和右子节点,将左子节点定义为 $(R(t_1))$, R

$(t_1.c)$), 右子节点定义为 $(R(t_r), R(t_r.c))$, 其中 t_1 和 t_r 为 WHERE 子句包含的数据表名, 且数据表 t_1 和 t_r 具有连接关系, $t_1.c$ 和 $t_r.c$ 为 WHERE 子句包含的列名, 且分别属于数据表 t_1 和 t_r , $R(t_1)$ 为数据表 t_1 的特征向量, $R(t_r)$ 为数据表 t_r 的特征向量, $R(t_1.c)$ 为列 $t_1.c$ 的特征向量, $R(t_r.c)$ 为列 $t_r.c$ 的特征向量;

[0178] 在本实施例, 步骤 S22 具体为: 对向量树 AT 输入的左子节点为 $(R(L), R(L.or))$, 右子节点为 $(R(O), R(O.or))$, 其中 $R(L)$ 为数据表 Lineitem 的特征向量, $R(O)$ 为数据表 Orders 的特征向量, $R(L.or)$ 为数据表 Lineitem 包含的列 orderkey 的特征向量, $R(O.or)$ 为数据表 Orders 包含的列 orderkey 的特征向量;

[0179] S23、对向量树 AT 输入左子节点 $(R(t_1), R(t_1.c))$ 和右子节点 $(R(t_r), R(t_r.c))$, 生成连接树的特征向量, 计算公式为:

$$[0180] \quad e = \text{sigmod}(W_e * [R(t_1.c), R(t_r.c)] + Y_e * [R(t_1), R(t_r)] + b_e)$$

$$[0181] \quad o = \text{tanh}(W_o * [R(t_1.c), R(t_r.c)] + Y_o * [R(t_1), R(t_r)] + b_o)$$

$$[0182] \quad h = e * o$$

[0183] 其中, $\text{sigmod}()$ 和 $\text{tanh}()$ 为激活函数, e 为对左右子节点的信息融合更新的程度, o 为左右子节点被提取的信息, h 为连接树的特征向量, W_e 和 Y_e 为对信息融合更新的权重参数, b_e 为对信息融合更新的偏置参数, W_o 和 Y_o 为对信息进行提取的权重参数, b_o 为对信息进行提取的偏置参数。

[0184] 在本实施例, 步骤 S23 具体为: 对向量树 AT 输入左子节点 $(R(L), R(L.or))$, 和右子节点 $(R(O), R(O.or))$, 生成包含数据表 Lineitem 和数据表 Orders 的连接树的特征向量 h_{L0} , 公式如下:

$$[0185] \quad e = \text{sigmod}(W_e * [R(L.or), R(O.or)] + Y_e * [R(L), R(O)] + b_e)$$

$$[0186] \quad o = \text{tanh}(W_o * [R(L.or), R(O.or)] + Y_o * [R(L), R(O)] + b_o)$$

$$[0187] \quad h_{L0} = e * o$$

[0188] 其中, $\text{sigmod}()$ 和 $\text{tanh}()$ 为激活函数, e 为对左右子节点的信息融合更新的程度, o 为左右子节点被提取的信息, h_{L0} 为包含数据表 Lineitem 和数据表 Orders 的连接树的特征向量。

[0189] S3、根据列、数据表、连接关系和连接树的特征向量, 设计部分连接计划模型 SP 来生成部分连接计划的特征向量, 进而生成下一时刻连接状态的特征向量;

[0190] 在本实施例中, 步骤 S3 具体为: 在初始时, 还没有任何数据表进行连接, 那么当前部分连接计划处于数据表无连接的形态, 为了便于举例子进行说明, 根据步骤 S2 将数据表 Lineitem 和数据表 Orders 进行连接而生成的连接树, 将包含数据表 Lineitem 和数据表 Orders 的连接树作为当前部分连接计划, 且 $H_1 = h_{L0}$, 那么在当前, 有两个数据表 Lineitem 和 Orders 已经连接, 再引入数据表 Supplier、Customer 作为例子, 设计部分连接计划模型 SP 将这些数据表连接在一起, 生成下一时刻部分连接计划的特征向量, 进而生成下一时刻连接状态的特征向量。

[0191] 步骤 S3 包括以下分步骤:

[0192] S31、根据列、数据表和连接树的特征向量, 构建部分连接计划模型 SP;

[0193] S32、根据部分连接计划模型 SP, 生成下一时刻部分连接计划的特征向量;

[0194] 所述步骤 S32 具体为:

[0195] 所述部分连接计划模型SP包括：第一类部分连接计划模型SP和第二类部分连接计划模型SP；

[0196] 所述第一类部分连接计划模型SP将当前部分连接计划与单个数据表进行连接，且当前部分连接计划中某一个数据表与单个数据表具有连接关系；

[0197] 将所述第一类部分连接计划模型SP的输入定义为第一类子节点和当前子节点，将第一类子节点定义为 $(R(t), R(t.c))$ ，其中， t 为WHERE子句包含的数据表名， $t.c$ 为WHERE子句包含的列名，且属于数据表 t ， $R(t)$ 为数据表 t 的特征向量， $R(t.c)$ 为列 $t.c$ 的特征向量；将当前子节点定义为 $(H_i, R(t_H.c))$ ，其中， H 为部分连接计划的特征向量， H_i 为当前部分连接计划的特征向量， t_H 为当前部分连接计划包含的数据表名， $t_H.c$ 为数据表 t_H 包含的列名，且数据表 t 与数据表 t_H 具有连接关系；

[0198] 对所述第一类部分连接计划模型SP输入第一类子节点和当前子节点，生成下一时刻部分连接计划的特征向量，计算公式为：

$$[0199] \quad g_i = \text{sigmod}(W_g * [R(t_H.c), R(t.c)] + Y_g * [H_i, R(t)] + b_g)$$

$$[0200] \quad e_i = \text{sigmod}(W_e * [R(t_H.c), R(t.c)] + Y_e * [H_i, R(t)] + b_e)$$

$$[0201] \quad o_i = \tanh(W_o * [R(t_H.c), R(t.c)] + Y_o * [g_i * H_i, R(t)] + b_o)$$

$$[0202] \quad H_{i+1} = (1 - e_i) * H_i + e_i * o_i$$

[0203] 其中， $\text{sigmod}()$ 和 $\tanh()$ 为激活函数， g_i 为当前子节点的信息没有被保留的程度， e_i 为对当前子节点和第一类子节点的信息提取的程度， o_i 为第一类子节点和当前子节点被保留的信息， H_{i+1} 为下一时刻部分连接计划的特征向量， W_g 和 Y_g 为对信息没有进行保留的权重参数， b_g 为对信息没有进行保留的偏置参数， W_e 和 Y_e 为对信息进行提取的权重参数， b_e 为对信息进行提取的偏置参数， W_o 和 Y_o 为对信息进行保留的权重参数， b_o 为对信息进行保留的偏置参数， W_e 、 W_o 、 Y_e 、 Y_o 、 b_e 、 b_o 与向量树AT共享；

[0204] 在本实施例中，步骤S32中第一类部分连接计划模型SP具体为：第一类部分连接计划模型SP的连接过程如图4，第一类部分连接计划模型SP的输入为第一类子节点 $(R(S), R(S.su))$ 和当前子节点 $(H_i, R(L.su))$ ，其中， $R(S)$ 为数据表Supplier的特征向量， $R(S.su)$ 为数据表Supplier包含的列suppkey的特征向量； H_i 为当前部分连接计划的特征向量， $R(L.su)$ 为数据表Lineitem包含的列suppkey的特征向量，且数据表Lineitem与数据表Supplier具有连接关系；对第一类部分连接计划模型SP输入第一类子节点和当前子节点，生成下一时刻部分连接计划的特征向量，计算公式为：

$$[0205] \quad g_i = \text{sigmod}(W_g * [R(L.su), R(S.su)] + Y_g * [H_i, R(S)] + b_g)$$

$$[0206] \quad e_i = \text{sigmod}(W_e * [R(L.su), R(S.su)] + Y_e * [H_i, R(S)] + b_e)$$

$$[0207] \quad o_i = \tanh(W_o * [R(L.su), R(S.su)] + Y_o * [g_i * H_i, R(S)] + b_o)$$

$$[0208] \quad H_{i+1} = (1 - e_i) * H_i + e_i * o_i$$

[0209] 其中， $\text{sigmod}()$ 和 $\tanh()$ 为激活函数， g_i 为对当前子节点的信息不保留的程度， e_i 为对当前子节点和第一类子节点的信息提取的程度， o_i 为第一类子节点和当前子节点被保留的信息， H_{i+1} 为下一时刻部分连接计划的特征向量， W_g 、 W_e 、 W_o 、 Y_g 、 Y_e 、 Y_o 为部分连接计划模型SP的权重参数， b_g 、 b_e 、 b_o 为部分连接计划模型SP的偏置参数， W_e 、 W_o 、 Y_e 、 Y_o 、 b_e 、 b_o 与向量树AT共享；

[0210] 所述第二类部分连接计划模型SP将当前部分连接计划与连接树进行连接，且当前

部分连接计划中某一个数据表与连接树中某一个数据表具有连接关系；

[0211] 将所述第二类部分连接计划模型SP的输入定义为第二类子节点和当前子节点，将第二类子节点定义为 $(h, R(t_h.c))$ ，其中 h 为连接树的特征向量， t_h 为连接树包含的数据表名， $t_h.c$ 为数据表 t_h 包含的列名；将当前子节点定义为 $(H_i, R(t_H.c))$ ，其中 H 为部分连接计划的特征向量， H_i 为当前部分连接计划的特征向量， t_H 为当前部分连接计划包含的数据表名， $t_H.c$ 为数据表 t_H 包含的列名，数据表 t_h 与数据表 t_H 具有连接关系；

[0212] 对所述第二类部分连接计划模型SP输入第二类子节点和当前子节点，生成下一时刻部分连接计划的特征向量，计算公式为：

$$[0213] \quad g_i = \text{sigmoid}(W_g * [R(t_H.c), R(t_h.c)] + Y_g * [H_i, h] + b_g)$$

$$[0214] \quad e_i = \text{sigmoid}(W_e * [R(t_H.c), R(t_h.c)] + Y_e * [H_i, h] + b_e)$$

$$[0215] \quad o_i = \tanh(W_o * [R(t_H.c), R(t_h.c)] + Y_o * [g_i * H_i, h] + b_o)$$

$$[0216] \quad H_{i+1} = (1 - e_i) * H_i + e_i * o_i$$

[0217] 其中， $\text{sigmoid}()$ 和 $\text{tanh}()$ 为激活函数， g_i 为对当前子节点的信息没有被保留的程度， e_i 为当前子节点与第二类子节点的信息提取的程度， o_i 为第二类子节点和当前子节点被保留的信息， H_{i+1} 为下一时刻部分连接计划的特征向量， W_g 和 Y_g 为对信息没有进行保留的权重参数， b_g 为对信息没有进行保留的偏置参数， W_e 和 Y_e 为对信息进行提取的权重参数， b_e 为对信息进行提取的偏置参数， W_o 和 Y_o 为对信息进行保留的权重参数， b_o 为对信息进行保留的偏置参数， W_e 、 W_o 、 Y_e 、 Y_o 、 b_e 、 b_o 与向量树AT共享；

[0218] 在本实施例中，步骤S32中第二类部分连接计划模型SP具体为：第二类部分连接计划模型SP的连接过程如图5，第二类部分连接计划模型SP的输入为第二类子节点 $(h_{SC}, R(S.su))$ 和当前子节点 $(H_i, R(L.su))$ ，其中 h_{SC} 为包含数据表Supplier和数据表Customer的连接树的特征向量， $R(S.su)$ 为数据表Supplier包含的列suppkey的特征向量； H_i 为当前部分连接计划的特征向量， $R(L.su)$ 为数据表Lineitem包含的列suppkey的特征向量，且数据表Lineitem与数据表Supplier具有连接关系；对第二类部分连接计划模型SP输入第二类子节点和当前子节点，生成下一时刻部分连接计划的特征向量，计算公式为：

$$[0219] \quad g_i = \text{sigmoid}(W_g * [R(L.su), R(S.su)] + Y_g * [H_i, h_{SC}] + b_g)$$

$$[0220] \quad e_i = \text{sigmoid}(W_e * [R(L.su), R(S.su)] + Y_e * [H_i, h_{SC}] + b_e)$$

$$[0221] \quad o_i = \tanh(W_o * [R(L.su), R(S.su)] + Y_o * [g_i * H_i, h_{SC}] + b_o)$$

$$[0222] \quad H_{i+1} = (1 - e_i) * H_i + e_i * o_i$$

[0223] 其中， $\text{sigmoid}()$ 和 $\text{tanh}()$ 为激活函数， g_i 为对当前子节点的信息不保留的程度， e_i 为对当前子节点与第二类子节点的信息提取的程度， o_i 为第二类子节点和当前子节点被保留的信息， H_{i+1} 为下一时刻部分连接计划的特征向量， W_g 、 W_e 、 W_o 、 Y_g 、 Y_e 、 Y_o 为部分连接计划模型SP的权重参数， b_g 、 b_e 、 b_o 为部分连接计划模型SP的偏置参数， W_e 、 W_o 、 Y_e 、 Y_o 、 b_e 、 b_o 与向量树AT共享；

[0224] S33、根据连接关系的特征向量和下一时刻部分连接计划的特征向量生成下一时刻连接状态的特征向量，计算公式为：

$$[0225] \quad R(s) = R(U) \oplus H_{i+1}$$

[0226] 其中， $R(s)$ 为下一时刻连接状态 s 的特征向量， $R(U)$ 为连接关系的特征向量， H_{i+1} 为下一时刻部分连接计划的特征向量。

[0227] S4、根据下一时刻连接状态的特征向量,构建深度强化学习模型J,并结合部分连接计划模型SP与向量树AT,生成数据表的最优连接顺序。

[0228] 本实施例中,深度强化学习模型J架构如图6所示。为了便于对如何生成数据表的最优连接顺序的整体流程进行说明,那么对深度强化学习模型J的训练过程就不进行举例说明,默认深度强化学习模型J的训练已经完成,对深度强化学习模型J的所有参数更新到最优。采用步骤S3中第一类部分连接计划模型SP所生成下一时刻部分连接计划的特征向量作为例子进行说明。首先将下一时刻部分连接计划的特征向量来重新代替新的当前部分连接计划1的特征向量,新的当前部分连接计划1如图8所示,表示已经将数据表Lineitem、数据表Orders和数据表Supplier进行连接,接下来根据估计价值神经网络得到连接动作,如选择数据表Supplier和数据表Customer进行连接(此处仅作假设,真实情况需要估计价值神经网络预测得到连接动作),然后根据选择的数据表Customer不存在于当前部分连接计划1,而数据表Supplier存在于当前部分连接计划1,那么采用第一类部分连接计划模型SP将当前部分连接计划1与数据表Customer进行连接,生成新的下一时刻部分连接计划的特征向量。再将新的下一时刻部分连接计划的特征向量来重新代替新的当前部分连接计划2的特征向量,新的当前部分连接计划2如图9所示,表示已经将数据表Lineitem、数据表Orders、数据表Supplier和数据表Customer进行连接,再根据估计价值神经网络得到连接动作,如选择数据表Nation和数据表Region进行连接(此处仅作假设,真实情况需要估计价值神经网络预测得到连接动作),然后根据选择的两个数据表均不存在于当前部分连接计划2,但数据表Nation与当前部分连接计划2包含的数据表Supplier具有连接关系,那么采用第二类部分连接计划模型SP将当前部分连接计划2与包含数据表Nation和数据表Region的连接树进行连接,实现将图7的SQL语句中WHERE子句包含的全部数据表都连接起来,生成数据表的最优连接顺序如图10所示。

[0229] 步骤S4中深度强化学习模型J包括:估计价值神经网络和目标价值神经网络;所述估计价值神经网络和目标价值神经网络结构相同,均包括:输入层、卷积层、价值函数层、优势函数层和全连接层;所述输入层的输入端作为估计价值神经网络或目标价值神经网络的输入端,其输出端与卷积层的输入端连接;所述卷积层的输出端分别与价值函数层的输入端和优势函数层的输入端连接;所述价值函数层的输出端和优势函数层的输出端均与全连接层的输入端连接;所述全连接层的输出端作为估计价值神经网络或目标价值神经网络的输出端。

[0230] 所述步骤S4包括以下分步骤:

[0231] S401、采用下一时刻部分连接计划的特征向量来重新代替新的当前部分连接计划的特征向量,采用下一时刻连接状态的特征向量来重新代替新的当前连接状态的特征向量;

[0232] S402、将新的当前连接状态的特征向量输入估计价值神经网络,得到当前连接状态下所有连接动作对应的估计价值;

[0233] S403、采用 ϵ -贪婪法对当前连接状态下所有连接动作对应的估计价值进行筛选,得到当前连接状态下最小估计价值和当前连接状态下最小估计价值对应的连接动作,其中,连接动作表示在WHERE子句包含的数据表中选择两个具有连接关系的数据表进行连接;

[0234] 步骤S403中当前连接状态下最小估计价值和当前连接状态下最小估计价值对应

的连接动作的计算公式为：

$$[0235] \quad P_{est} = \min(P_{est}(R(s), A, w_{est}, \alpha, \beta))$$

$$[0236] \quad A^{\min}(R(s), w_{est}, \alpha, \beta) = \operatorname{argmin} P_{est}(R(s), A, w_{est}, \alpha, \beta)$$

[0237] 其中, P_{est} 为当前连接状态下最小估计价值, \min 为最小值函数, $P_{est}()$ 为估计价值神经网络, $R(s)$ 为新的当前连接状态 s 的特征向量, A 为连接动作, w_{est} 为估计价值神经网络的权重参数, α 为价值函数层的状态参数, β 为优势函数层的状态参数, $A^{\min}(R(s), w_{est}, \alpha, \beta)$ 为当前连接状态下最小估计价值对应的连接动作, argmin 为反函数;

[0238] S404、根据选择的两个具有连接关系的数据表和新的当前部分连接计划的特征向量, 结合向量树AT和部分连接计划模型SP, 来生成新的下一时刻部分连接计划的特征向量;

[0239] 步骤S404具体为:

[0240] 当两个具有连接关系的数据表有且只有一个数据表不存在于当前部分连接计划中, 采用第一类部分连接计划模型SP将这个数据表与当前部分连接计划进行连接, 生成新的下一时刻部分连接计划的特征向量; 当这两个数据表均不存在于当前部分连接计划中, 但这两个数据表中某一个数据表与当前部分连接计划包含的某一个数据表具有连接关系时, 采用向量树AT将所选择的两个具有连接关系的数据表进行连接, 得到新的连接树, 然后采用第二类部分连接计划模型SP将当前部分连接计划与新的连接树进行连接, 生成新的下一时刻部分连接计划的特征向量。

[0241] S405、根据步骤S404所得新的下一时刻部分连接计划的特征向量, 采用步骤S33的计算公式来生成新的下一时刻连接状态的特征向量;

[0242] S406、根据S405所得新的下一时刻连接状态的特征向量生成估计奖励;

[0243] 所述步骤S406中生成估计奖励的计算公式为:

$$[0244] \quad R = \operatorname{sigmod}(W_R * R(s)_{new})$$

[0245] 其中, R 为估计奖励, $\operatorname{sigmod}()$ 为激活函数, $R(s)_{new}$ 为新的下一时刻连接状态 s 的特征向量, W_R 为计算估计奖励的参数;

[0246] S407、将步骤S401所得新的当前连接状态的特征向量、步骤S403所得最小估计价值对应的连接动作、步骤S405所得新的下一时刻连接状态的特征向量和步骤S406所得估计奖励组成单个样本放入训练集合ST中, 得到训练集合ST;

[0247] S408、根据步骤S405所得新的下一时刻连接状态的特征向量和步骤S406所得估计奖励, 计算目标估计价值;

[0248] 所述步骤S408中计算目标估计价值需要使用下一时刻连接状态下最小估计价值对应的连接动作, 计算公式为:

$$[0249] \quad A^{\min}(R(s)_{new}, w_{est}, \alpha, \beta) = \operatorname{argmin} P_{est}(R(s)_{new}, A, w_{est}, \alpha, \beta)$$

[0250] 其中, $A^{\min}(R(s)_{new}, w_{est}, \alpha, \beta)$ 为下一时刻连接状态下最小估计价值对应的连接动作, $R(s)_{new}$ 为新的下一时刻连接状态 s 的特征向量, w_{est} 为估计价值神经网络的权重参数, α 为价值函数层的状态参数, β 为优势函数层的状态参数, argmin 为反函数, $P_{est}()$ 为估计价值神经网络, A 为连接动作;

[0251] 所述步骤S408中目标估计价值的计算公式为:

$$[0252] \quad P_{target} = R + \gamma * P_{target}(R(s)_{new}, A^{\min}(R(s)_{new}, w_{est}, \alpha, \beta), w_{target})$$

[0253] 其中, P_{target} 为目标估计价值, R 为步骤S405所得估计奖励, γ 为衰减程度因子, 取

值范围为0到1, $P_{\text{target}}()$ 为目标价值神经网络, $R(s)_{\text{new}}$ 为新的下一时刻连接状态s的特征向量, $A^{\text{min}}(R(s)_{\text{new}}, w_{\text{est}}, \alpha, \beta)$ 为下一时刻连接状态下最小估计价值对应的连接动作, w_{est} 为估计价值神经网络的权重参数, α 为价值函数层的状态参数, β 为优势函数层的状态参数, w_{target} 为目标价值神经网络的目标权重参数;

[0254] S409、根据步骤S403所得最小估计价值和步骤S408所得目标估计价值,从训练集合ST中采样z个样本,构建均方差损失函数;

[0255] 所述步骤S409构建均方差损失函数需要计算样本的采样优先级、样本的采样概率和样本的权重;

[0256] 所述样本的采样优先级的计算公式为:

$$[0257] \quad \&_r = |P_{\text{target}} - P_{\text{est}}|$$

[0258] 其中,r为样本序号,&_r为第r个样本的采样优先级,|·|为绝对值函数, P_{target} 为目标估计价值, P_{est} 为最小估计价值;

[0259] 所述样本的采样概率的计算公式为:

$$[0260] \quad P(r) = \&_r / (\&_1 + \dots + \&_z), r \leq z$$

[0261] 其中,r为样本序号,z为样本数量, $P(r)$ 为第r个样本的采样概率,&_r为第r个样本的采样优先级,&₁, ..., &_z为第1个样本的采样优先级至第z个样本的采样优先级;

[0262] 所述样本的权重的计算公式为:

$$[0263] \quad w_r = (P(r))^{-z} / \max(w_1, \dots, w_z)$$

[0264] 其中,r为样本序号,z为样本数量, w_r 为第r个样本的权重, $P(r)$ 为第r个样本的采样概率,max()为最大值函数, w_1, \dots, w_z 为第1个样本的权重至第z个样本的权重;

[0265] 所述步骤S409中均方差损失函数的计算公式为:

$$[0266] \quad L = 1/z \sum_{r=1}^z w_r * (P_{\text{target}} - P_{\text{est}})^2$$

[0267] 其中,r为样本序号,z为样本数量,L为均方差损失函数值, Σ 为求和函数, w_r 为第r个样本的权重, P_{target} 为目标估计价值, P_{est} 为最小估计价值。

[0268] S410、根据均方差损失函数,通过神经网络的梯度反向传播来更新估计价值神经网络的所有参数,并且每隔额定时间,将估计价值神经网络的所有参数的参数值传递给目标价值神经网络,进而更新目标价值神经网络的所有参数;

[0269] S411、在更新目标价值神经网络所有参数的基础上,根据步骤S404所得新的下一时刻部分连接计划的特征向量和步骤S405所得新的下一时刻连接状态的特征向量,依次循环步骤S401、S402、S403、S404、S405、S406、S407、S408、S409和S410,直到完整连接状态,实现对深度强化学习模型J的所有参数完成一轮更新;

[0270] S412、采用多个SQL语句对深度强化学习模型J的所有参数完成多轮更新时,得到深度强化学习模型J的最优参数,生成数据表的最优连接顺序。

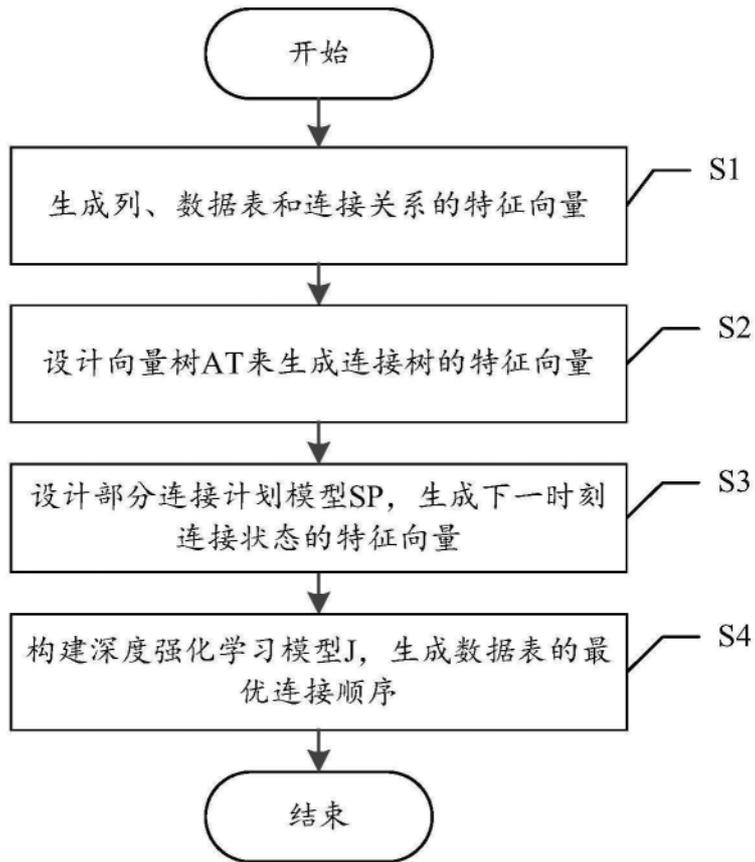


图1

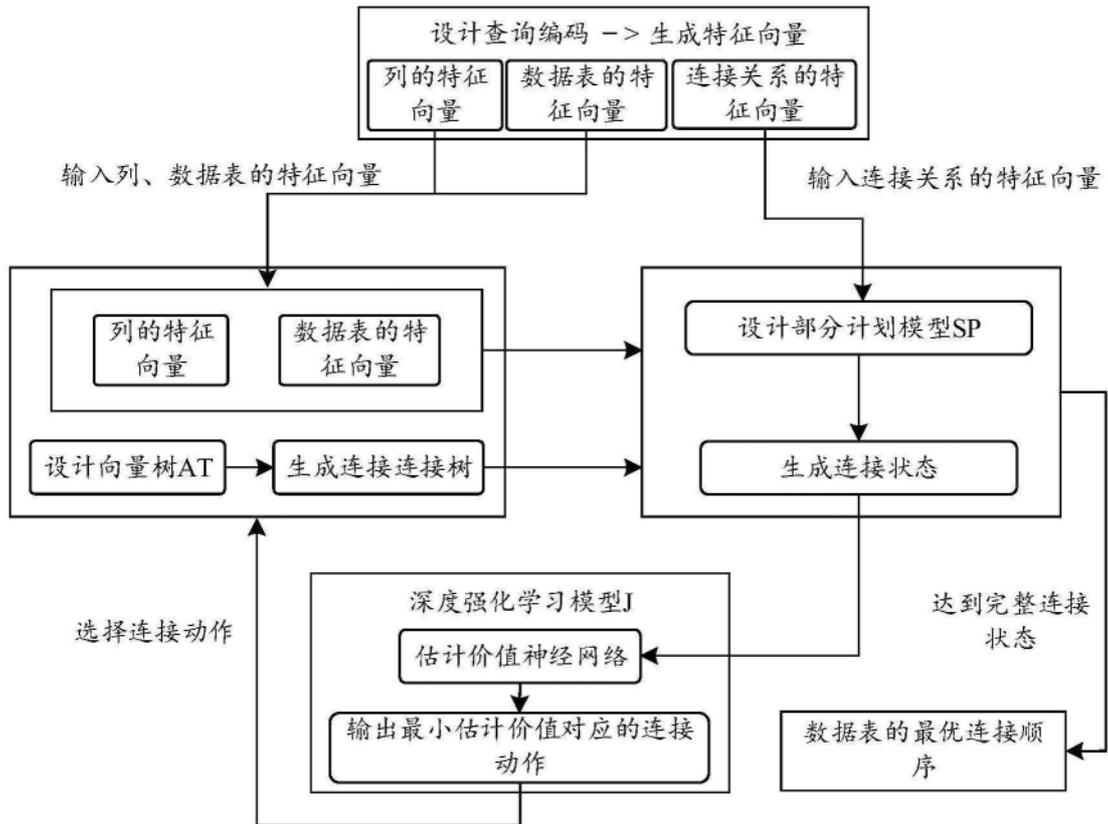


图2

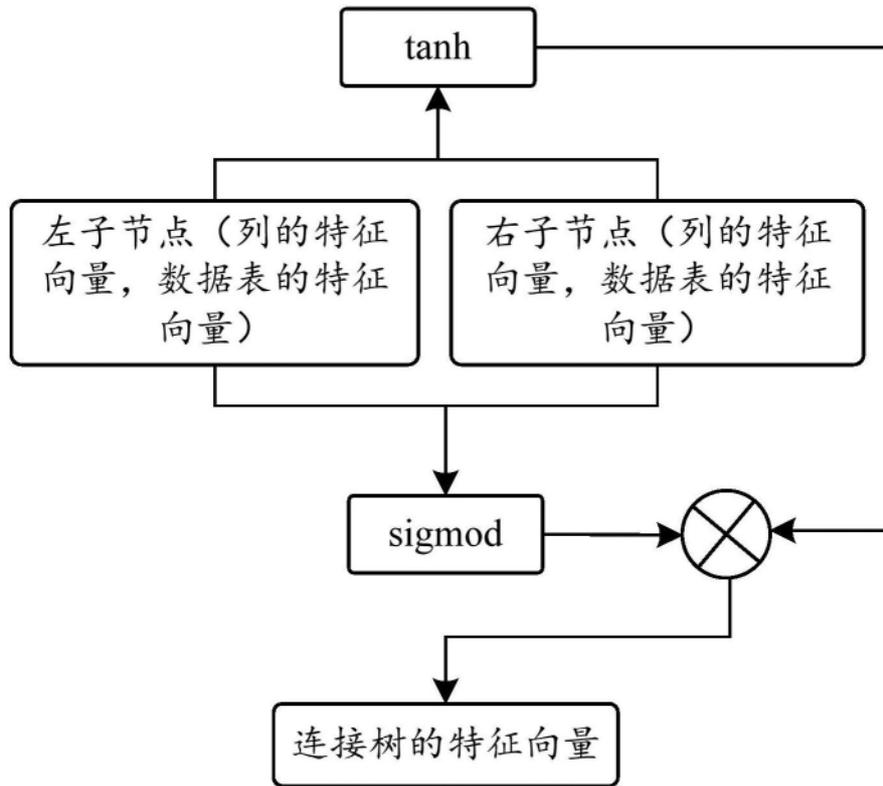
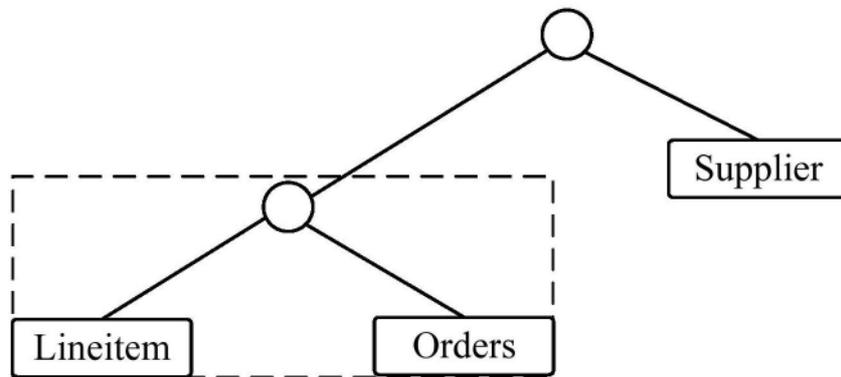


图3



当前部分连接计划

○ 表示“连接”

图4

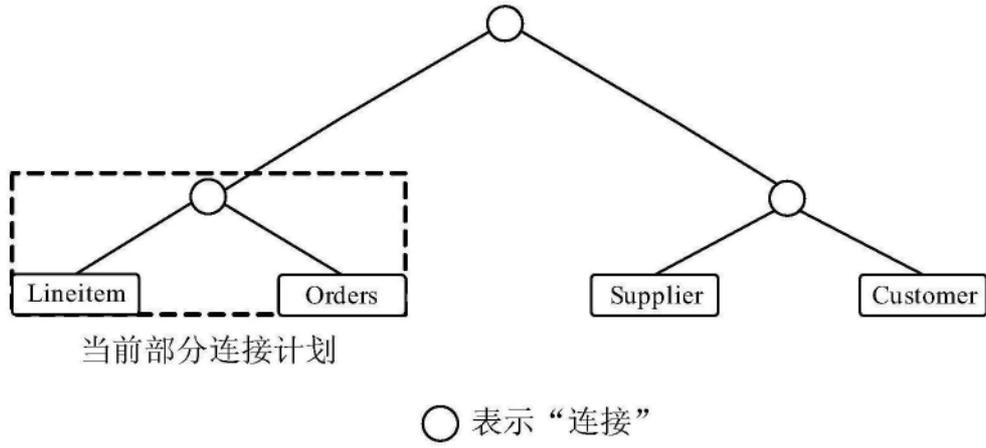


图5

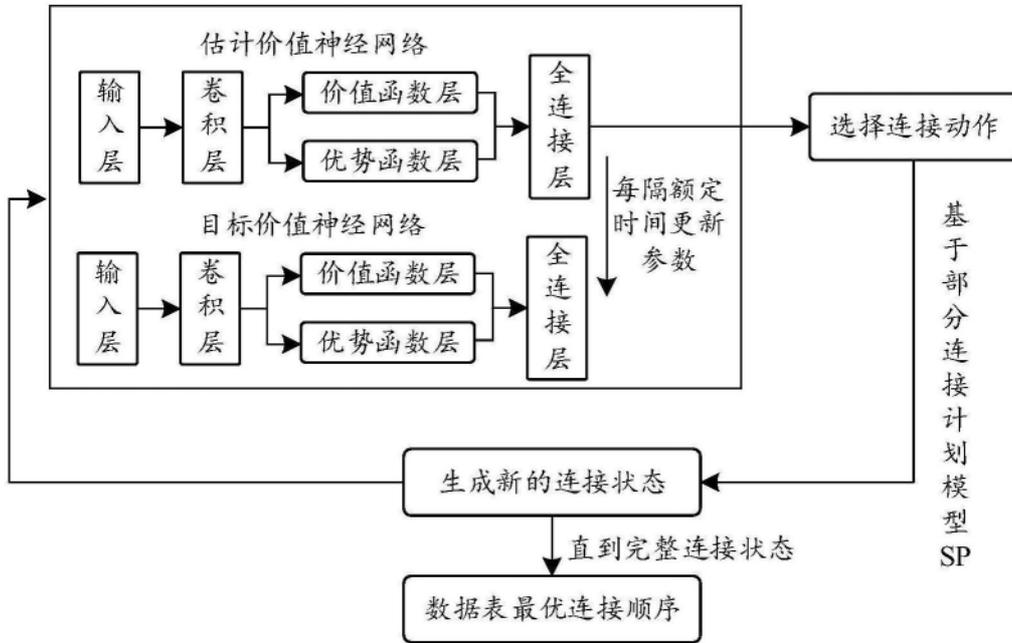
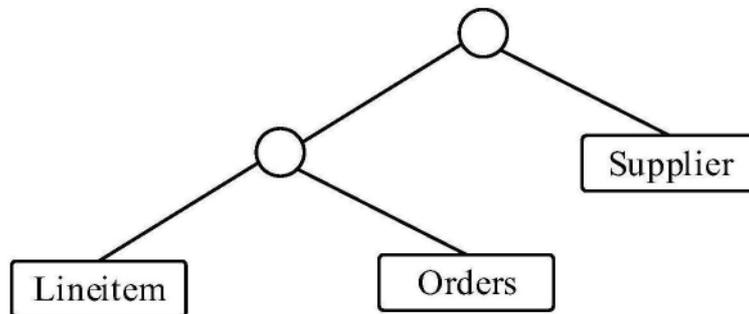


图6

SQL语句例子:

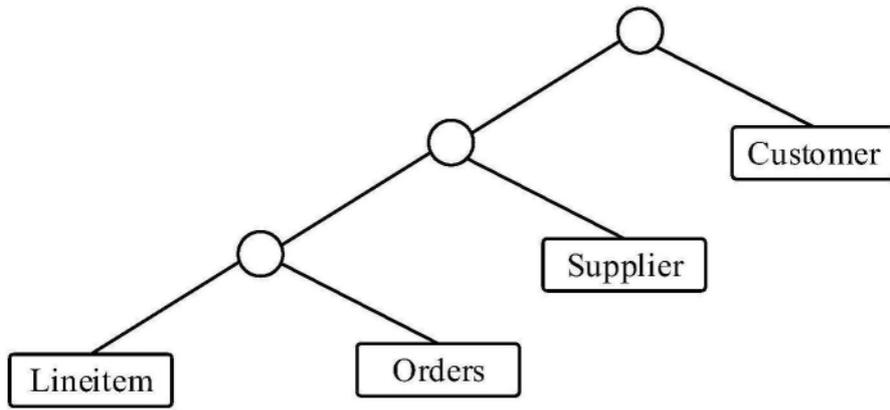
```
select
n_name,l_extendedprice,l_discount
from
customer,orders,lineitem,supplier,nation,region //六表连接
where
c_custkey = o_custkey
and l_orderkey = o_orderkey
and l_suppkey = s_suppkey
and c_nationkey = s_nationkey
and s_nationkey = n_nationkey
and n_regionkey = r_regionkey
and r_name = '[REGION]' //指定地区, 在TPC-H标准指定的范围内随机选择
and o_orderdate >= date '[DATE]' //DATE是从1993年到1997年中随机选择的一年的1月1日
and o_orderdate < date '[DATE]' + interval '1' year
```

图7



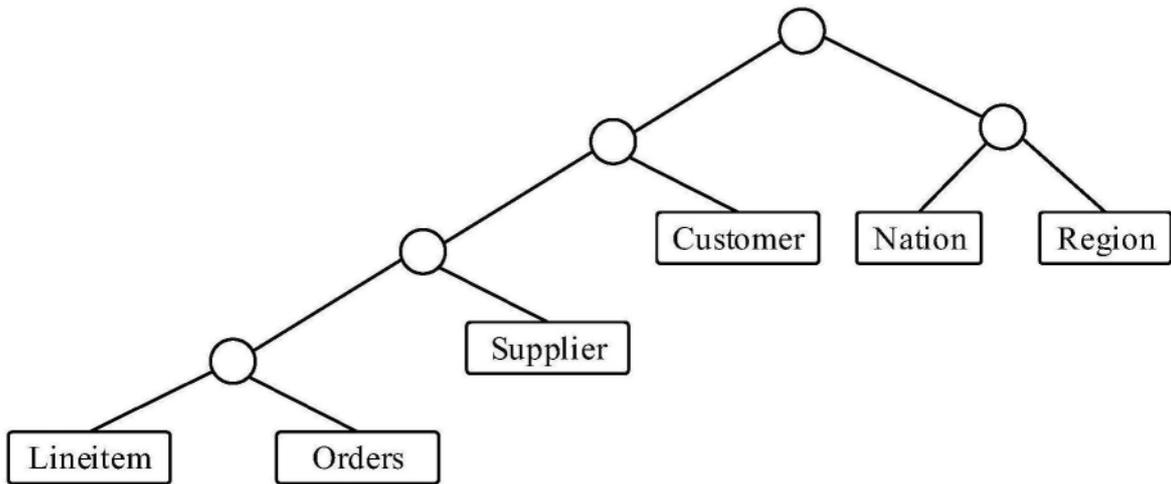
○ 表示“连接”

图8



○ 表示“连接”

图9



○ 表示“连接”

图10