(12) 按照专利合作条约所公布的国际申请

(19) 世界知识产权组织
国 际 局

(43) 国际公布日
2020 年 12 月 17 日 (17.12.2020)

WIPO | PCT

(10) 国际公布号
WO 2020/248440 A1

(51) 国际专利分类号:
G06N 20/00 (2019.01) *G06N 3/08* (2006.01)

(21) 国际申请号: PCT/CN2019/108984

(22) 国际申请日: 2019 年 9 月 29 日 (29.09.2019)

(25) 申请语言: 中文

(26) 公布语言: 中文

(30) 优先权:
201910511173.2 2019年6月13日 (13.06.2019) CN

(71) 申请人: 苏州浪潮智能科技有限公司 (INSPUR SUZHOU INTELLIGENT TECHNOLOGY CO., LTD) [CN/CN]; 中国江苏省苏州市官浦路 1 号 9幢, Jiangsu 215000 (CN)。

(72) 发明人: 李峰 (LI, Feng); 中国江苏省苏州市官浦路 1 号 9 幢, Jiangsu 215000 (CN)。 刘红丽 (LIU, Hongli); 中国江苏省苏州市官浦路 1 号 9 幢, Jiangsu 215000 (CN)。 刘宏刚 (LIU, Honggang); 中国江苏省苏州市官浦路 1 号 9幢, Jiangsu 215000 (CN)。

(74) 代理人: 北京集佳知识产权代理有限公司 (UNITALEN ATTORNEYS AT LAW); 中国北京市朝阳区建国门外大街 22 号赛特广场 7层, Beijing 100004 (CN)。

(81) 指定国(除另有指明，要求每一种可提供的国家保护): AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DJ, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IR, IS,

(54) **Title:** MACHINE LEARNING METHOD AND APPARATUS

(54) **发明名称:** 一种机器学习方法与装置

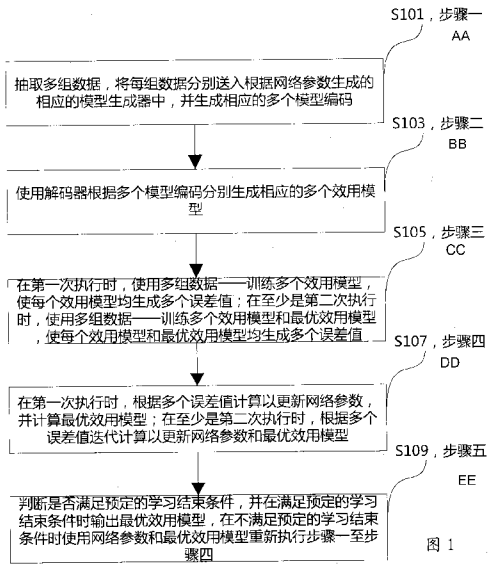

图 1

S101 Extract multiple sets of data, feed each set of data into a corresponding model generator generated according to a network parameter, and generate multiple corresponding model codes

S103 Use a decoder to generate multiple corresponding utility models according to the multiple model codes

S105 During a first time being executed, use the multiple sets of data one by one to train the multiple utility models, causing each utility model to generate multiple error values; during the second and following times being executed, use the multiple sets of data one by one to train the multiple utility models and an optimum utility model, causing each utility model and the optimum utility model to generate multiple error values

S107 During a first time being executed, calculate according to the multiple error values so as to update a network parameter, and calculate an optimum utility model; during the second and following times being executed, iteratively calculate according to the multiple error values so as to update the network parameter and the optimum utility model

S109 Determine whether a preset learning end condition has been satisfied, output the optimum utility model when the preset learning end condition is satisfied, and when the preset learning end condition is not satisfied, use the network parameter and the optimum utility model to execute steps 1 to 4 again

AA Step 1
BB Step 2
CC Step 3
DD Step 4
EE Step 5

(57) **Abstract:** A machine learning method and an apparatus, comprising: extracting multiple sets of data, feeding each set of data into a corresponding model generator generated according to a network parameter, and generating multiple corresponding model codes; using a decoder to generate multiple corresponding utility models according to the multiple model codes; using the multiple sets of data one by one to train the multiple utility models and an optimum utility model, causing each utility model and the optimum utility model to generate multiple error values; iteratively calculating according to the multiple error values so as to update the network parameter

WO 2020/248440 A1

[见续页]

JO, JP, KE, KG, KH, KN, KP, KR, KW, KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SA, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW。

(84) 指定国(除另有指明，要求每一种可提供的地区保护)：ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, ST, SZ, TZ, UG, ZM, ZW), 欧亚 (AM, AZ, BY, KG, KZ, RU, TJ, TM), 欧洲 (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, KM, ML, MR, NE, SN, TD, TG)。

本国际公布:

— 包括国际检索报告(条约第21条(3))。

and the optimum utility model; determining whether a preset learning end condition has been satisfied, outputting the optimum utility model when the preset learning end condition is satisfied, and when the preset learning end condition is not satisfied, using the network parameter and the optimum utility model to execute the described steps again. This technical solution can free up labor, improve work efficiency, and lower costs.

(57) 摘要：一种机器学习方法与装置，包括：通过抽取多组数据，将每组数据分别送入根据网络参数生成的相应的模型生成器中，并生成相应的多个模型编码；使用解码器根据多个模型编码分别生成相应的多个效用模型；使用多组数据一一训练多个效用模型和最优效用模型，使每个效用模型和最优效用模型均生成多个误差值；根据多个误差值迭代计算以更新网络参数和最优效用模型；判断是否满足预定的学习结束条件，并在满足预定的学习结束条件时输出最优效用模型，在不满足预定的学习结束条件时使用网络参数和最优效用模型重新执行上述步骤的技术方案，能够解放人工，提高工作效率，降低成本。

一种机器学习方法与装置

本申请要求于 2019 年 6 月 13 日提交中国专利局、申请号为201910511173.2、发明名称为“一种机器学习方法与装置”的中国专利申请的优先权，其全部内容通过引用结合在本申请中。

技术领域

本发明涉及机器学习领域，并且更具体地，特别是涉及一种机器学习方法与装置，以及相应的计算机。

背景技术

基于神经网络的人工智能算法在人工智能技术兴起后吸引了众多学者和产业应用的兴趣。然而，由于实际生产生活中面临的任务不尽相同，针对不同的任务目前需要人为地设计解决问题用的模型和算法。另一方面，深度神经网络、卷积网络等神经网络结构因在各项任务中的突出表现而引起了学界和工业界的浓厚兴趣。但是，面向网络结构搜索的自动机器学习算法由于计算量庞大，系统设计复杂，始终限制了这一技术在算法研究人员和算法工程师中的普及和应用。

现有技术使用 AutoML（auto machine learning）来输入样本数据，通过模型生成器生成适用于特定任务的机器学习模型。AutoML 的机器学习模型以生成决策树等传统手段为主，但随着深度学习的兴起，面向神经网络结构的 AutoML 开始逐渐受到关注。此类算法的核心是构造面向神经网络结构的搜索空间（Neural architecture space, NAS），通过优化搜索策略寻找适用于特定任务的模型结构。由于所给定的任务和样本数据不同，这一类 AutoML 算法只能针对特定任务进行构建而不能一般化。

针对现有技术中针对神经网络结构的 AutoML 使用的人工设计方式耗时高、依赖个人经验、而自动化机器学习方法效率低耗时长的问题，目前尚未有有效的解决方案。

发明内容

有鉴于此，本发明实施例的目的在于提出一种机器学习方法与装置，能够针对一般化的不同任务或不同类型的任务构建结构模型来进行自动化机器学习，解放人工，提高工作效率，降低成本。

基于上述目的，本发明实施例的一方面提供了一种机器学习方法，包括以下步骤：

步骤一：抽取多组数据，将每组数据分别送入根据网络参数生成的相应的模型生成器中，并生成相应的多个模型编码；

步骤二：使用解码器根据多个模型编码分别生成相应的多个效用模型；

步骤三：在第一次执行时，使用多组数据一一训练多个效用模型，使每个效用模型均生成多个误差值；在至少是第二次执行时，使用多组数据一一训练多个效用模型和最优效用模型，使每个效用模型和最优效用模型均生成多个误差值；

步骤四：在第一次执行时，根据多个误差值计算以更新网络参数，并计算最优效用模型；在至少是第二次执行时，根据多个误差值迭代计算以更新网络参数和最优效用模型；

步骤五：判断是否满足预定的学习结束条件，并在满足预定的学习结束条件时输出最优效用模型，在不满足预定的学习结束条件时使用网络参数和最优效用模型重新执行步骤一至步骤四。

在一些实施方式中，使用多组数据一一训练多个效用模型和最优效用模型包括：

将多组数据一一输入多个效用模型和最优效用模型，获得多个预测输出；

根据多个预测输出和由用户指定的与其相对应的样本标签确定交叉熵损失函数；

使用交叉熵损失函数来训练多个效用模型和最优效用模型。

在一些实施方式中，根据多个误差值迭代计算以更新网络参数和最优效用模型包括：

根据多个误差值确定每个效用模型和最优效用模型的总体误差；

根据多个误差值和多个总体误差确定每个效用模型和最优效用模型的

精度损失;

根据多个精度损失确定每个效用模型和最优效用模型相对于多组数据的收益函数;

根据多个收益函数更新网络参数和最优效用模型。

在一些实施方式中，根据多个收益函数更新网络参数和最优效用模型包括:

确定每个效用模型和最优效用模型的计算复杂度损失;

根据多个计算复杂度损失和多个收益函数确定每个效用模型和最优效用模型的强化学习损失;

根据多个强化学习损失更新网络参数;

根据多个计算复杂度损失和多个收益函数确定每个效用模型和最优效用模型的辅助矩阵;

根据多个辅助矩阵和多个收益函数确定每个效用模型和最优效用模型的偏移值;

根据多个偏移值更新最优效用模型。

在一些实施方式中，确定每个效用模型和最优效用模型的计算复杂度损失包括:

根据与每个效用模型和最优效用模型相对应的多个模型编码的各指示位数量和计算量分别确定每个效用模型和最优效用模型的当前计算量;

根据多个当前计算量确定每个效用模型和最优效用模型的额外计算量;

根据多个当前计算量和多个额外计算量确定每个效用模型和最优效用模型的计算复杂度损失。

在一些实施方式中，指示位包括通常位置指示位、规约位置指示位、和空层指示位，并且通常位置指示位和规约位置指示位具有相同或不同的计算量，空层指示位具有零计算量。

在一些实施方式中，在第一次执行时，使用并行处理方法针对多个效用模型同时执行步骤一、步骤二、和步骤三；在至少是第二次执行时，使用并行处理方法针对多个效用模型和最优效用模型同时执行步骤一、步骤

二、和步骤三。

在一些实施方式中，预定的学习结束条件包括以下至少之一：学习达到指定的迭代次数；多个模型编码在指定的迭代次数内不发生改变；最优效用模型的收益函数达到预定阈值。

基于上述目的，本发明实施例的另一方面提供了一种机器学习装置，包括：

处理器；和

存储器，存储有处理器可运行的程序代码，所述程序代码在被运行时执行上述的方法。

基于上述目的，本发明实施例的另一方面提供了一种计算机，包括：

存储和计算单元集群，设置于计算机底层，用于执行并行计算任务；

多线程分布式任务调度模块，设置于存储和计算单元集群的上一层，用于接收并行计算任务，并根据配置参数将不同的并行计算任务选择性地下发到存储和计算单元集群中的不同单元以优化计算性能；

接口模块，设置于多线程分布式任务调度模块的上一层，封装有多个可被调用的深度学习框架，用于维护深度学习框架的更新和兼容性，其中深度学习框架用于将分类和训练任务转化为并行计算任务并下发到多线程分布式任务调度模块；

模型类，设置于接口模块的上一层，包括模型生成器组和效用模型组，其中模型生成器组根据数据样本生成模型编码，效用模型组根据模型编码来对数据样本执行分类任务；

模型训练类，设置于接口模块的上一层，包括模型生成器训练组和效用模型训练组，其中模型生成器训练组使用反向传播方法对模型生成器组执行训练任务，效用模型训练组使用强化 Q 学习方法对效用模型组执行训练任务；

误差计算模块，设置于模型类和模型训练类的上一层，根据效用模型组的数据样本分类输出使用不同类型的方法分别计算多种误差，并根据误差在效用模型组中确定一个最优效用模型；

输出模块，用于判断是否满足预定的学习结束条件，并据此输出当前

的所述最优效用模型。

本发明具有以下有益技术效果：本发明实施例提供的机器学习方法与装置，通过抽取多组数据，将每组数据分别送入根据网络参数生成的相应的模型生成器中，并生成相应的多个模型编码；使用解码器根据多个模型编码分别生成相应的多个效用模型；使用多组数据一一训练多个效用模型和最优效用模型，使每个效用模型和最优效用模型均生成多个误差值；根据多个误差值迭代计算以更新网络参数和最优效用模型；判断是否满足预定的学习结束条件，并在满足预定的学习结束条件时输出最优效用模型，在不满足预定的学习结束条件时使用网络参数和最优效用模型重新执行上述步骤的技术方案，能够针对一般化的不同任务或不同类型的任务构建结构模型来进行自动化机器学习，通过自动化机器学习来解放人工，提高工作效率，降低成本。

附图说明

为了更清楚地说明本发明实施例或现有技术中的技术方案，下面将对实施例或现有技术描述中所需要使用的附图作简单地介绍，显而易见地，下面描述中的附图仅仅是本发明的一些实施例，对于本领域普通技术人员来讲，在不付出创造性劳动的前提下，还可以根据这些附图获得其他的实施例。

图 1 为本发明提供的机器学习方法的流程示意图；

图 2 为本发明提供的机器学习方法的迭代训练流程图；

图 3 为本发明提供的机器学习方法的模型输出流程图；

图 4 为本发明提供的机器学习方法的误差计算示意图；

图 5 为本发明提供的机器学习方法的计算复杂度矩阵图；

图 6 为本发明提供的机器学习方法的并行计算示意图；

图 7 为本发明提供的机器学习方法的学习效果折线图；

图 8 为本发明提供的计算机的层级结构图。

具体实施方式

为使本发明的目的、技术方案和优点更加清楚明白，以下结合具体实施例，并参照附图，对本发明实施例进一步详细说明。

需要说明的是，本发明实施例中所有使用“第一”和“第二”的表述均是为了区分两个相同名称非相同的实体或者非相同的参量，可见“第一”、“第二”仅为了表述的方便，不应理解为对本发明实施例的限定，后续实施例对此不再一一说明。

基于上述目的，本发明实施例的第一个方面，提出了一种能够针对一般化的不同任务或不同类型的任务构建结构模型的机器学习方法的实施例。图 1 示出的是本发明提供的机器学习方法的实施例的流程示意图。

所述机器学习方法，包括以下步骤:

步骤 S101，抽取多组数据，将每组数据分别送入根据网络参数生成的相应的模型生成器中，并生成相应的多个模型编码;

步骤 S103，使用解码器根据多个模型编码分别生成相应的多个效用模型;

步骤 S105，在第一次执行时，使用多组数据一一训练多个效用模型，使每个效用模型均生成多个误差值；在至少是第二次执行时，使用多组数据一一训练多个效用模型和最优效用模型，使每个效用模型和最优效用模型均生成多个误差值;

步骤 S107，在第一次执行时，根据多个误差值计算以更新网络参数，并计算最优效用模型；在至少是第二次执行时，根据多个误差值迭代计算以更新网络参数和最优效用模型;

步骤 S109，判断是否满足预定的学习结束条件，并在满足预定的学习结束条件时输出最优效用模型，在不满足预定的学习结束条件时使用网络参数和最优效用模型重新执行步骤一至步骤四。

在一些实施方式中，使用多组数据一一训练多个效用模型和最优效用模型包括:

将多组数据一一输入多个效用模型和最优效用模型，获得多个预测输出;

根据多个预测输出和由用户指定的与其相对应的样本标签确定交叉熵

损失函数;

使用交叉熵损失函数来训练多个效用模型和最优效用模型。

在一些实施方式中，根据多个误差值迭代计算以更新网络参数和最优效用模型包括:

根据多个误差值确定每个效用模型和最优效用模型的总体误差;

根据多个误差值和多个总体误差确定每个效用模型和最优效用模型的精度损失;

根据多个精度损失确定每个效用模型和最优效用模型相对于多组数据的收益函数;

根据多个收益函数更新网络参数和最优效用模型。

在一些实施方式中，根据多个收益函数更新网络参数和最优效用模型包括:

确定每个效用模型和最优效用模型的计算复杂度损失;

根据多个计算复杂度损失和多个收益函数确定每个效用模型和最优效用模型的强化学习损失;

根据多个强化学习损失更新网络参数;

根据多个计算复杂度损失和多个收益函数确定每个效用模型和最优效用模型的辅助矩阵;

根据多个辅助矩阵和多个收益函数确定每个效用模型和最优效用模型的偏移值;

根据多个偏移值更新最优效用模型。

在一些实施方式中，确定每个效用模型和最优效用模型的计算复杂度损失包括:

根据与每个效用模型和最优效用模型相对应的多个模型编码的各指示位数量和计算量分别确定每个效用模型和最优效用模型的当前计算量;

根据多个当前计算量确定每个效用模型和最优效用模型的额外计算量;

根据多个当前计算量和多个额外计算量确定每个效用模型和最优效用模型的计算复杂度损失。

在一些实施方式中，指示位包括通常位置指示位、规约位置指示位、和空层指示位，并且通常位置指示位和规约位置指示位具有相同或不同的计算量，空层指示位具有零计算量。

在一些实施方式中，在第一次执行时，使用并行处理方法针对多个效用模型同时执行步骤一、步骤二、和步骤三；在至少是第二次执行时，使用并行处理方法针对多个效用模型和最优效用模型同时执行步骤一、步骤二、和步骤三。

在一些实施方式中，预定的学习结束条件包括以下至少之一：学习达到指定的迭代次数；多个模型编码在指定的迭代次数内不发生改变；最优效用模型的收益函数达到预定阈值。

下面根据具体实施例进一步阐述本发明的详细实施方式。

本发明实施例的迭代训练流程参见图 2。首先，抽取 m 组 Batch 数据（即前述的数据组），将 m 组 Batch 数据分别送入 m 个 Model Generator（即前述的模型生成器，以下简称 MG）中，每个 MG 输出一个 model code（即前述的模型编码）。Model Decoder（即前述的解码器）接收 model code 后按照解码规则将其解析为一个 utility model（即前述的效用模型，以下简称 UM），用 m 组 Batch 数据训练每一个 UM。为了保证 MG 的学习收敛性，本发明实施例将当前 UM 中 loss（即前述的误差）值最小的一个模型（被称为 Hb Model）保留到下一代中，与之后生成的模型做比较。

Batch 数据经过所有 UM 后会产生 m 个 loss 值。考虑到每个模型每次仅采用了一批采样数据进行训练，为了保持模型对整体样本集的适用性，本发明实施例需要对模型的 loss 值进行累计，得到loss_{total}。由于每一次采样后，MG 生成的 UM 不同，因此这个loss_{total}表征的不是 UM 而是所对应的 MG 的性能。

通过综合计算 UM 的 loss 值和loss_{total}可以得到具有最好精度性能的 MG。精度损失采用如下公式计算：

$$\mathrm{Acc} = \lambda \cdot \mathrm{loss} + (1-\lambda)\mathrm{loss}_{total} \quad (1)$$

其中λ是权重参数，精度损失越小则则对应的模型越理想。为了平衡计算精度和计算负载，本发明实施例将模型的复杂度损失 f 作为惩罚函数纳

入考量。复杂度损失包含以下两个部分:

$$f = f_{comp} + f_{cost} \tag{2}$$

其中，f_{comp}表示当前 MG 所生成 UM 的计算量，f_{cost}表示如果采用其他 UM，所需要额外付出的计算量。综合上述精度和计算性能指标，每个 MG 在给定的样本和参数条件下，都可以得到一个评分。为了方便表述，本发明实施例对任意一个 MG，令s_t: $(x_{sample}, \theta_{[MG]})$, a: utility model，并且使用$r_{t+1}$表示采用该 UM 所得到的收益，其数值与各个模型的 loss 和$loss_{total}$成反比，由此得$f_{cost} = f_{comp}(a_{t+1}) - f_{comp}$。基于上述符号系统，本发明实施例定义再当前样本和 MG 模型参数下，向任意 UM 学习（即下一次生成所选的 UM）的 Q 值为:

$$O(s_{[t+1]}, a_t) = r_{[t+1]}(s_{[t+1]}, a_t) - f(a_t) \tag{3}$$

$$Q_{new}(s_t, a_t) = (1-\alpha) \cdot Q_{old}(s_t, a_t) + \alpha[r_t + \gamma \cdot \max_a O(s_{t+1}, a)] \tag{4}$$

Q 值越大则对应的模型越理想。其中，O 为观测矩阵，表示在 t+1 时刻采用 t 时刻的模型进行预测所获得的收益和代价。对第 i 个 MG 可以确定作为学习对象的参考模型（即前述的最优效用模型）$a_i^* = \mathrm{argmax}_{a \in [a]} Q_{new}$。在对应的样本输入下，每个 MG 分别学习自己对应的参考模型，MG 对每一组 batch 数据的学习次数 epoch_MG 可根据 UM 的训练时间来设定以最大限度保证模型训练时间重叠。

处理完所有样本数据称为完成了一次搜索。当搜索次数达到用户指定的次数上限，或最后一批数据经过 MG 所产生的 Model code 在指定的学习代数内不发生改变时 AutoML 学习结束，如图 3 所示地输出经过解码和预训练后的 r 值最高的 UM。也可以选择继续对返回的 UM 进行进一步训练。

关于精度损失和复杂度损失的详细计算方法参见以下实施例。

对于一次采样得到的$M = m * batch$个样本数据，设第 i 个 UM 对 M 个样本的预测输出为$\hat{y}_i$，样本对应的标签为y_{lab}，并使用交叉熵来衡量分类问题模型的损失函数，则有:

$$loss_H = -\frac{1}{M}\sum_{(j=1)}^{M}\sum_{(k=1)}^{n} y_{(lab)}\log(\hat{y}_i) \quad (5)$$

如图 4 所示，$loss_H$用来训练 GM。当 UM 被训练过指定个 epoch 后，最后一次模型的输出记为$\overline{loss}^i$。对于第 K 次完成训练时，MG 所对应的模型的$loss_{total}$为：

$$loss_{total} = \frac{K-1}{K}(\overline{loss}^i + loss_{total}) \quad (6)$$

应注意上式是迭代更新公式而不是等式。此时精度损失为：

$$Acc = \lambda \cdot \overline{loss}^i + (1-\lambda)loss_{total} \quad (7)$$

需要注意，应先计算 Acc 的值，然后迭代更新$loss_{total}$；K=1 时，$loss_{total}$取 0。

模型的计算复杂度通常由 model code 决定，模型复杂度可规定如下：

Model Code 结构和复杂度定义		
含义	字符位	计算量
Normal_cell	10	$f_{comp,N}$
Reduce_cell	10	$f_{comp,R}$
Empty layer	10	0

Normal_cell 即前述的通常位置指示位，Reduce_cell 即前述的规约位置指示位，Empty layer 即前述的空层指示位。计算由多个 cell 执行，每个 cell 因其功能与运算方式不同而具有不同的计算复杂度。对于第 i 个 UM，其计算复杂度为：

$$f_{comp} = \sum I_N f_{comp,N} + \sum I_R f_{comp,R} \quad (8)$$

其中$f_{comp,N}$和$f_{comp,R}$可以提前指定或由 Normal cell 和 Reduce cell 计算

求得。上式中的 I 为 cell 指示位。根据上式可以计算出每一个 UM 当前的计算复杂度，则对于模型 i，复杂度损失此时为:

$$f^i = 2f^{ref}_{comp} - f^i_{comp} \tag{9}$$

图 5 示出的是此时的 f 矩阵。

关于网络参数更新的详细计算方法参见以下实施例。

首先，基于精度损失定义收益函数 r:

$$r_t = \frac{1}{Acc(s_t, a_t)} \tag{10}$$

收益函数 r 为在 t 时刻各 UM 对样本 的收益。Q 矩阵为行数随时间增长的二维矩阵，其列数等于 MG 的数量+1（多一个 Hb 模型）。为了更新 MG 的参数，本发明实施例将参考模型学习和 Q 值看作 MG 的奖励，并采用策略梯度法更新 MG 生成 cell 过程中做出不同选择的概率。

更新 MG 参数的核心问题是构建合理的损失函数，并通过最小化损失函数提高生成理想的 UM 的概率。本发明实施例把理想的 MG 视为能够最大化 um 模型的期望收益，即:

$$\max_{\theta_c} \mathbb{E}(P_{\theta_c}[R]) \tag{11}$$

其中[R]表示策略模型的收益，P_{θ_c}表示采用参数集θ_c生成当前 UM 的概率。将每一个 UM 看做对 MG 的一次采样，上式可以近似为以下形式:

$$\max_{\theta_c} \frac{1}{m} \sum_{k=1}^{m} \prod_{t=i}^{T} \frac{\sum I(t)}{m} R_k \tag{12}$$

其中 m 表示用于训练 MG 的一个 batch 中 UM 的数量，T 表示 UMde 编码长度。上式中，$\sum I(t)$表示对于所有 UM 的第 t 位编码与第 k 个 UM 的

第 t 位编码相同的模型数量，即对第 k 个模型统计每一位在 batch 中出现相同编码的频率。本发明实施例采用独立分布假设对 MG 生成编码的联合概率分布进行近似。R_k表示第 k 个 MG 获得的奖励，使用观测矩阵带入R_k，则上式可以改写为：

$$\max_{\theta_c}: \frac{1}{m}\sum_{k=1}^{m}\sum_{t=1}^{T}\log\frac{\sum I(t)}{m}[r_k(s_t,a_t)-\bar{f}^k] \tag{13}$$

由于 TensorFlow 自带的优化器只能向极小值方向优化，因此对优化函数取反，最终 MG 训练的优化函数为：

$$Loss_{RL} = -\frac{1}{m}\sum_{k=1}^{m}\sum_{t=1}^{T}\log\frac{\sum I(t)}{m}[r_k(s_t,a_t)-\bar{f}^k] \tag{14}$$

图 6 示出的是并行计算示意图。如图 6 所示，在 AutoML 系统训练阶段，MG 从 CPU 接受网络参数θ_c和 MG 个数 L，在 GPU0 中采用模型并行的方式构建 L 个结构相同、参数不同的 MG，共用同一个会话，但是输入和输出互不干扰。每个 MG 生成 m 个 UM 并计算误差和奖励。对每一个 MG 所创建的 m 个 UM，当完成 m 个 R 值计算后即打包并传入 Reward buffer（奖励缓冲）。CPU 依次从 reward buffer 中取出数据包，计算数据包所对应的 MG 的参数集合θ_c，然后更新相应参数。参数进行如下迭代更新：

$$\nabla\theta_c = \nabla\theta_c \cdot Loss_{RL} \tag{15}$$

实验证明了本发明实施例的有效性。本发明实施例将在 Cifar-10 图像数据集上生成图像分类模型作为测试实例展示本发明实施例的有益效果。

本发明实施例采用 RNN（循环神经网络）作为 GM，步长为 5，即每一次输出 5 个编码值来表示两个输入和各自的操作，以及这两个输入的合并方式。上述 5 个输入作为一组，重复五次构成的五组输出并合并为一个 cell。GM 生成两类不同的 cell 以及其堆叠方式的编码。解码器将根据预先给定的编码规则对照生成 UM。由于 cifar-10 数据集中的数据为彩色图片，因此对于接收输入数据的第一层网络，需要将输入通道数强制修改为

3。对 UM 的 loss 值计算采用交叉熵函数（见公式（5）），训练采用误差反传法，训练过程选取 SGD（随机最速下降）作为优化器。

当 UM 被训练过指定个 epoch 后，最后一次模型的输出记为$\overline{loss^i}$；对于第 K 次完成训练时，MG 所对应的模型的$loss_{total}$按照公式（6）计算；对于第 i 个 UM，其计算复杂度可根据公式（8）计算；根据公式（7）（9）-（14）计算出每个 MG 对应的 loss 值；并使用公式（15）更新相应参数。图 7 示出了采用强化学习迭代 29 次时，MG 的误差变化情况，其中每次迭代每个 UM 模型经过 10 个 epoch 训练。由图 7 可见，误差随迭代次数而大幅度降低。

从上述实施例可以看出，本发明实施例提供的机器学习方法，通过抽取多组数据，将每组数据分别送入根据网络参数生成的相应的模型生成器中，并生成相应的多个模型编码；使用解码器根据多个模型编码分别生成相应的多个效用模型；使用多组数据一一训练多个效用模型和最优效用模型，使每个效用模型和最优效用模型均生成多个误差值；根据多个误差值迭代计算以更新网络参数和最优效用模型；判断是否满足预定的学习结束条件，并在满足预定的学习结束条件时输出最优效用模型，在不满足预定的学习结束条件时使用网络参数和最优效用模型重新执行上述步骤的技术方案，能够针对一般化的不同任务或不同类型的任务构建结构模型来进行自动化机器学习，通过自动化机器学习来解放人工，提高工作效率，降低成本。

需要特别指出的是，上述机器学习方法的各个实施例中的各个步骤均可以相互交叉、替换、增加、删减，因此，这些合理的排列组合变换之于机器学习方法也应当属于本发明的保护范围，并且不应将本发明的保护范围局限在所述实施例之上。

基于上述目的，本发明实施例的第二个方面，提出了一种能够针对一般化的不同任务或不同类型的任务构建结构模型的机器学习装置的实施例。所述装置包括:

处理器；和

存储器，存储有处理器可运行的程序代码，所述程序代码在被运行时执行如上述的方法。其中，程序代码还包括以下部分:

网络结构搜索模块，用于抽取多组数据，将每组数据分别送入根据网络参数生成的相应的模型生成器中，并生成相应的多个模型编码;

解码模块，用于对多个模型编码使用解码器分别生成相应的多个效用模型;

计算模块，用于使用多组数据一一训练多个效用模型和已有的最优效用模型，使每个效用模型和最优效用模型均生成多个误差值，根据多个误差值迭代计算更新的网络参数和更新的最优效用模型，并使用更新的网络参数和更新的最优效用模型覆盖已有的网络参数和已有的最优效用模型调用网络结构搜索模块和解码模块重新执行上述步骤;

其中机器学习装置使用上述模块执行上述方法，并在满足预定的学习结束条件时输出当前的最优效用模型。

从上述实施例可以看出，本发明实施例提供的机器学习装置，通过抽取多组数据，将每组数据分别送入根据网络参数生成的相应的模型生成器中，并生成相应的多个模型编码；使用解码器根据多个模型编码分别生成相应的多个效用模型；使用多组数据一一训练多个效用模型和最优效用模型，使每个效用模型和最优效用模型均生成多个误差值；根据多个误差值迭代计算以更新网络参数和最优效用模型；判断是否满足预定的学习结束条件，并在满足预定的学习结束条件时输出最优效用模型，在不满足预定的学习结束条件时使用网络参数和最优效用模型重新执行上述步骤的技术方案，能够针对一般化的不同任务或不同类型的任务构建结构模型来进行自动化机器学习，通过自动化机器学习来解放人工，提高工作效率，降低成本。

需要特别指出的是，上述机器学习装置的实施例采用了所述机器学习方法的实施例来具体说明各模块的工作过程，本领域技术人员能够很容易

想到，将这些模块应用到所述机器学习方法的其他实施例中。当然，由于所述机器学习方法实施例中的各个步骤均可以相互交叉、替换、增加、删减，因此，这些合理的排列组合变换之于所述机器学习装置也应当属于本发明的保护范围，并且不应将本发明的保护范围局限在所述实施例之上。

基于上述目的，本发明实施例的第三个方面，提出了一种能够针对一般化的不同任务或不同类型的任务构建结构模型的计算机的实施例。计算机包括:

存储和计算单元集群，设置于计算机底层，用于执行并行计算任务;

多线程分布式任务调度模块，设置于存储和计算单元集群的上一层，用于接收并行计算任务，并根据配置参数将不同的并行计算任务选择性地下发到存储和计算单元集群中的不同单元以优化计算性能;

接口模块，设置于多线程分布式任务调度模块的上一层，封装有多个可被调用的深度学习框架，用于维护深度学习框架的更新和兼容性，其中深度学习框架用于将分类和训练任务转化为并行计算任务并下发到多线程分布式任务调度模块;

模型类，设置于接口模块的上一层，包括模型生成器组和效用模型组，其中模型生成器组根据数据样本生成模型编码，效用模型组根据模型编码来对数据样本执行分类任务;

模型训练类，设置于接口模块的上一层，包括模型生成器训练组和效用模型训练组，其中模型生成器训练组使用反向传播方法对模型生成器组执行训练任务，效用模型训练组使用强化Q学习方法对效用模型组执行训练任务;

误差计算模块，设置于模型类和模型训练类的上一层，根据效用模型组的数据样本分类输出使用不同类型的方法分别计算多种误差，并根据误差在效用模型组中确定一个最优效用模型;

输出模块，用于判断是否满足预定的学习结束条件，并据此输出当前

的所述最优效用模型。

如图 8 所示，计算机底层存储和计算资源的硬件或虚拟资源，其上构建有任务调度模块。任务调度模块负责接收并行计算任务，通过给定的参数设置将其合理分配给底层计算资源。任务调度模块负责优化计算性能。在其上是目前主流的深度学习框架。为了避免由于深度学习框架更新迭代导致大量的代码维护工作，以及维持多框架的兼容性，本发明实施例用接口模块对所用到的操作进行封装。后续所有用到的操作均采用封装后的函数进行实现。

在定义了专用的接口模块后，采用重写的层操作构建模型类。模型类有两个派生，两个派生继承模型类中的基础属性和方法（如模型参数数量、模型图绘制、超参列表、推理训练方法等），一个是 GM，接收输入的样本来生成 model code 和 Cell；另一个是 UM，接收输入的 Model Code 和样本来构建完成给定分类任务的模型。其中 model code 用于构造 UM，样本用于训练 UM。MG 和 UM 都需要被训练，但由于 MG 需要采用 R-QL（强化 Q 学习）方法训练，而 UM 采用 BP（反向传播）方法训练，本发明实施例构造模型训练类来包含不同的训练方法，并实例化不同的训练器。

虽然不同模型训练过程中都需要接收标签信息来计算不同类型的 loss 值，但是优化器的输入数据是最终计算出来的 loss 值。本发明实施例把 loss 值计算单独构建成一个模块，在其中封装不同 loss 计算方式（如交叉熵、均方差等）和 loss 的定义（如强化 Q 学习中生成指定参考模型的概率）。

误差计算模块接收 UM 的输出和样本的标签并计算 loss、$loss_{total}$ 以和 Q 值。其中 loss 用于训练 UM，当模型达到搜索过程所指定的训练次数后，loss 将被用于计算 $loss_{total}$ 和 Q。后两者用于更新 MG。

从上述实施例可以看出，本发明实施例提供的计算机，通过抽取多组数据，将每组数据分别送入根据网络参数生成的相应的模型生成器中，并

生成相应的多个模型编码；使用解码器根据多个模型编码分别生成相应的多个效用模型；使用多组数据一一训练多个效用模型和最优效用模型，使每个效用模型和最优效用模型均生成多个误差值；根据多个误差值迭代计算以更新网络参数和最优效用模型；判断是否满足预定的学习结束条件，并在满足预定的学习结束条件时输出最优效用模型，在不满足预定的学习结束条件时使用网络参数和最优效用模型重新执行上述步骤的技术方案，能够针对一般化的不同任务或不同类型的任务构建结构模型来进行自动化机器学习，通过自动化机器学习来解放人工，提高工作效率，降低成本。

需要特别指出的是，上述计算机的实施例采用了所述机器学习方法的实施例来具体说明各模块的工作过程，本领域技术人员能够很容易想到，将这些模块应用到所述机器学习方法的其他实施例中。当然，由于所述机器学习方法实施例中的各个步骤均可以相互交叉、替换、增加、删减，因此，这些合理的排列组合变换之于所述计算机也应当属于本发明的保护范围，并且不应将本发明的保护范围局限在所述实施例之上。

以上是本发明公开的示例性实施例，但是应当注意，在不背离权利要求限定的本发明实施例公开的范围的前提下，可以进行多种改变和修改。根据这里描述的公开实施例的方法权利要求的功能、步骤和/或动作不需以任何特定顺序执行。此外，尽管本发明实施例公开的元素可以以个体形式描述或要求，但除非明确限制为单数，也可以理解为多个。

应当理解的是，在本文中使用的，除非上下文清楚地支持例外情况，单数形式“一个”旨在也包括复数形式。还应当理解的是，在本文中使用的“和/或”是指包括一个或者一个以上相关联地列出的项目的任意和所有可能组合。本发明实施例公开实施例序号仅仅为了描述，不代表实施例的优劣。

所属领域的普通技术人员应当理解：以上任何实施例的讨论仅为示例性的，并非旨在暗示本发明实施例公开的范围（包括权利要求）被限于这些例子；在本发明实施例的思路下，以上实施例或者不同实施例中的技术特征之间也可以进行组合，并存在如上所述的本发明实施例的不同方面的

许多其它变化，为了简明它们没有在细节中提供。因此，凡在本发明实施例的精神和原则之内，所做的任何省略、修改、等同替换、改进等，均应包含在本发明实施例的保护范围之内。

5

权 利 要 求

1. 一种机器学习方法，其特征在于，包括以下步骤:

步骤一：抽取多组数据，将每组所述数据分别送入根据网络参数生成的相应的模型生成器中，并生成相应的多个模型编码;

步骤二：使用解码器根据多个所述模型编码分别生成相应的多个效用模型;

步骤三：在第一次执行时，使用多组所述数据一一训练多个所述效用模型，使每个所述效用模型均生成多个误差值；在至少是第二次执行时，使用多组所述数据一一训练多个所述效用模型和所述最优效用模型，使每个所述效用模型和所述最优效用模型均生成多个所述误差值;

步骤四：在第一次执行时，根据多个所述误差值计算以更新所述网络参数，并计算最优效用模型；在至少是第二次执行时，根据多个所述误差值迭代计算以更新所述网络参数和所述最优效用模型;

步骤五：判断是否满足预定的学习结束条件，并在满足所述预定的学习结束条件时输出所述最优效用模型，在不满足所述预定的学习结束条件时使用所述网络参数和所述最优效用模型重新执行步骤一至步骤四。

2. 根据权利要求 1 所述的方法，其特征在于，使用多组所述数据一一训练多个所述效用模型和所述最优效用模型包括:

将多组所述数据一一输入多个所述效用模型和所述最优效用模型，获得多个预测输出;

根据多个预测输出和由用户指定的与其相对应的样本标签确定交叉熵损失函数;

使用所述交叉熵损失函数来训练多个所述效用模型和所述最优效用模型。

3. 根据权利要求 1 所述的方法，其特征在于，根据多个所述误差值迭代计算以更新所述网络参数和所述最优效用模型包括:

根据多个所述误差值确定每个所述效用模型和所述最优效用模型的总体误差;

根据多个所述误差值和多个所述总体误差确定每个所述效用模型和所述最优效用模型的精度损失;

根据多个所述精度损失确定每个所述效用模型和所述最优效用模型相对于多组所述数据的收益函数;

根据多个所述收益函数更新所述网络参数和所述最优效用模型。

4. 根据权利要求3所述的方法，其特征在于，根据多个所述收益函数更新所述网络参数和所述最优效用模型包括:

确定每个所述效用模型和所述最优效用模型的计算复杂度损失;

根据多个所述计算复杂度损失和多个所述收益函数确定每个所述效用模型和所述最优效用模型的强化学习损失;

根据多个所述强化学习损失更新所述网络参数;

根据多个所述计算复杂度损失和多个所述收益函数确定每个所述效用模型和所述最优效用模型的辅助矩阵;

根据多个所述辅助矩阵和多个所述收益函数确定每个所述效用模型和所述最优效用模型的偏移值;

根据多个所述偏移值更新所述最优效用模型。

5. 根据权利要求4所述的方法，其特征在于，确定每个所述效用模型和所述最优效用模型的所述计算复杂度损失包括:

根据与每个所述效用模型和所述最优效用模型相对应的多个所述模型编码的各指示位数量和计算量分别确定每个所述效用模型和所述最优效用模型的当前计算量;

根据多个所述当前计算量确定每个所述效用模型和所述最优效用模型的额外计算量;

根据多个所述当前计算量和多个所述额外计算量确定每个所述效用模型和所述最优效用模型的所述计算复杂度损失。

6. 根据权利要求5所述的方法，其特征在于，所述指示位包括通常位置指示位、规约位置指示位、和空层指示位，并且所述通常位置指示位和所述规约位置指示位具有相同或不同的计算量，所述空层指示位具有零计算量。

7. 根据权利要求1所述的方法，其特征在于，在第一次执行时，使用并行处理方法针对多个所述效用模型同时执行步骤一、步骤二、和步骤三；在至少是第二次执行时，使用并行处理方法针对多个所述效用模型和所述最优效用模型同时执行步骤一、步骤二、和步骤三。

8. 根据权利要求1所述的方法，其特征在于，所述预定的学习结束条件包括以下至少之一：学习达到指定的迭代次数；多个所述模型编码在指定的迭代次数内不发生改变；所述最优效用模型的收益函数达到预定阈值。

9. 一种机器学习装置，其特征在于，包括：

处理器；和

存储器，存储有处理器可运行的程序代码，所述程序代码在被运行时执行如权利要求1-8中任意一项所述的方法。

10. 一种计算机，其特征在于，包括：

存储和计算单元集群，设置于计算机底层，用于执行并行计算任务；

多线程分布式任务调度模块，设置于所述存储和计算单元集群的上一层，用于接收所述并行计算任务，并根据配置参数将不同的所述并行计算任务选择性地下发到所述存储和计算单元集群中的不同单元以优化计算性能；

接口模块，设置于所述多线程分布式任务调度模块的上一层，封装有多个可被调用的深度学习框架，用于维护所述深度学习框架的更新和兼容

性，其中所述深度学习框架用于将分类和训练任务转化为所述并行计算任务并下发到所述多线程分布式任务调度模块；

模型类，设置于所述接口模块的上一层，包括模型生成器组和效用模型组，其中所述模型生成器组根据数据样本生成模型编码，所述效用模型组根据所述模型编码来对所述数据样本执行所述分类任务；

模型训练类，设置于所述接口模块的上一层，包括模型生成器训练组和效用模型训练组，其中所述模型生成器训练组使用反向传播方法对所述模型生成器组执行所述训练任务，所述效用模型训练组使用强化Q学习方法对所述效用模型组执行所述训练任务；

误差计算模块，设置于所述模型类和所述模型训练类的上一层，根据所述效用模型组的所述数据样本分类输出使用不同类型的方法分别计算多种误差，并根据所述误差在所述效用模型组中确定一个最优效用模型；

输出模块，用于判断是否满足预定的学习结束条件，并据此输出当前的所述最优效用模型。

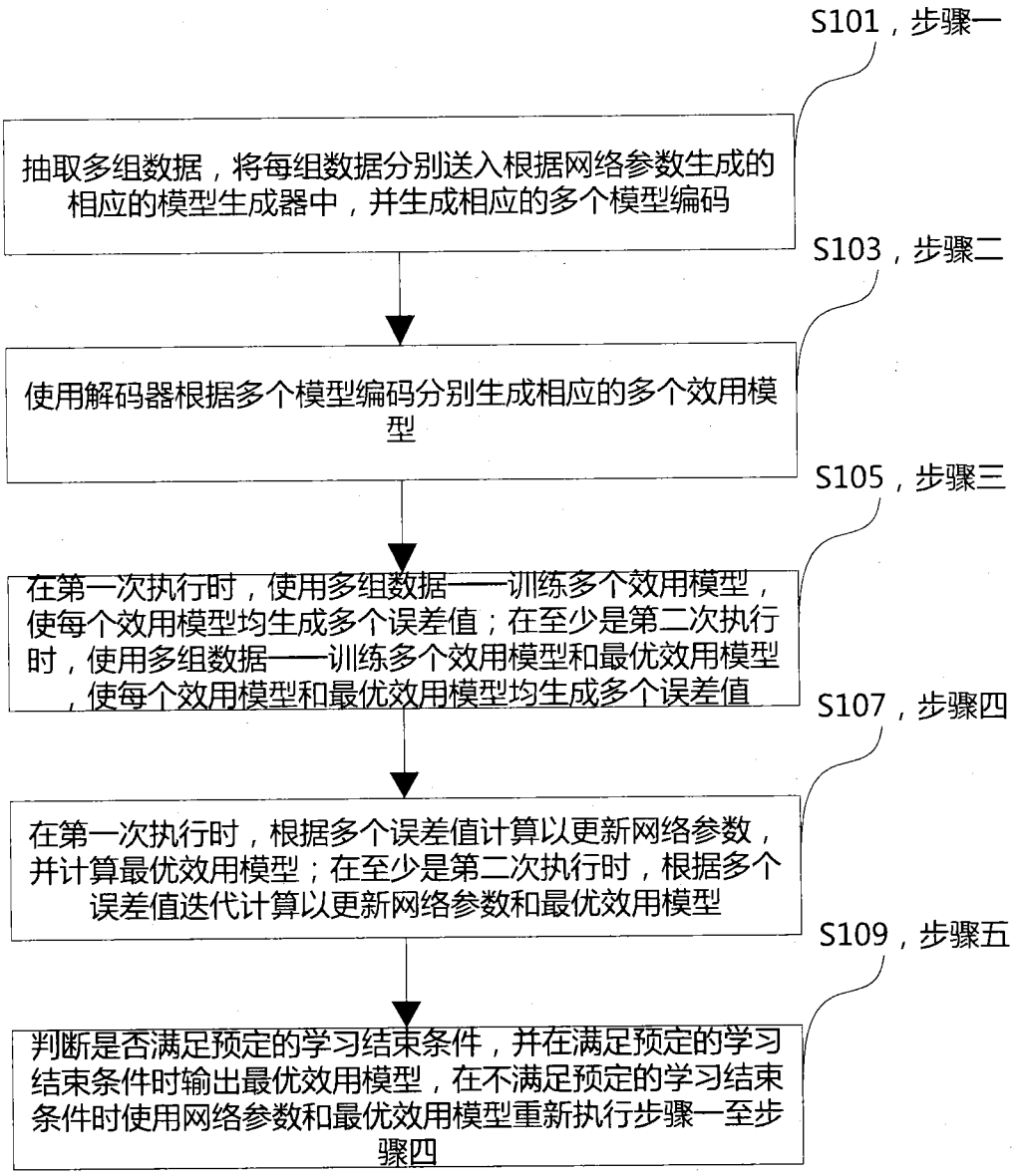
S101，步骤一
抽取多组数据，将每组数据分别送入根据网络参数生成的相应的模型生成器中，并生成相应的多个模型编码
S103，步骤二
使用解码器根据多个模型编码分别生成相应的多个效用模型
S105，步骤三
在第一次执行时，使用多组数据——训练多个效用模型，使每个效用模型均生成多个误差值；在至少是第二次执行时，使用多组数据——训练多个效用模型和最优效用模型，使每个效用模型和最优效用模型均生成多个误差值
S107，步骤四
在第一次执行时，根据多个误差值计算以更新网络参数，并计算最优效用模型；在至少是第二次执行时，根据多个误差值迭代计算以更新网络参数和最优效用模型
S109，步骤五
判断是否满足预定的学习结束条件，并在满足预定的学习结束条件时输出最优效用模型，在不满足预定的学习结束条件时使用网络参数和最优效用模型重新执行步骤一至步骤四

图 1

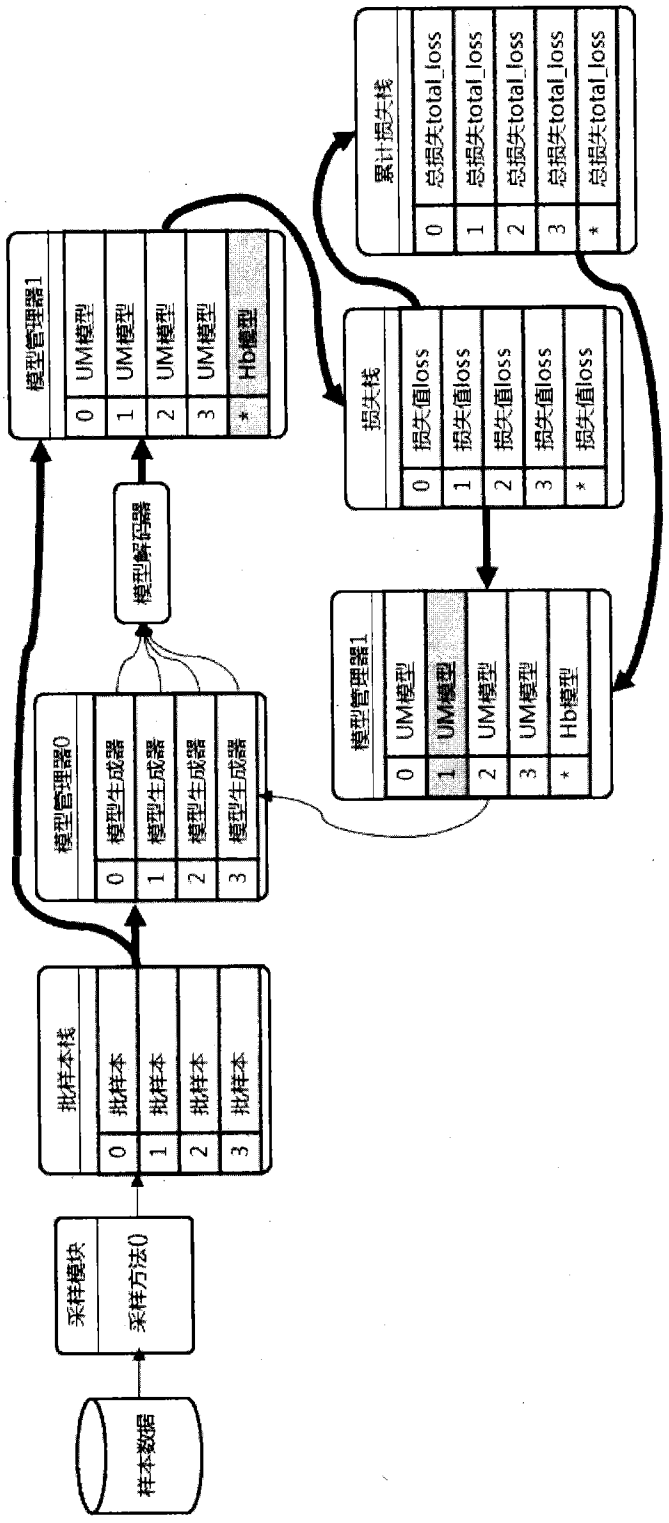
样本数据
采样模块
采样方法0
批样本栈
0 批样本
1 批样本
2 批样本
3 批样本
模型管理器0
0 模型生成器
1 模型生成器
2 模型生成器
3 模型生成器
模型解码器
模型管理器1
0 UM模型
1 UM模型
2 UM模型
3 UM模型
* Hb模型
损失栈
0 损失值loss
1 损失值loss
2 损失值loss
3 损失值loss
* 损失值loss
累计损失栈
0 总损失total_loss
1 总损失total_loss
2 总损失total_loss
3 总损失total_loss
* 总损失total_loss
模型管理器1
0 UM模型
1 UM模型
2 UM模型
3 UM模型
* Hb模型

图 2

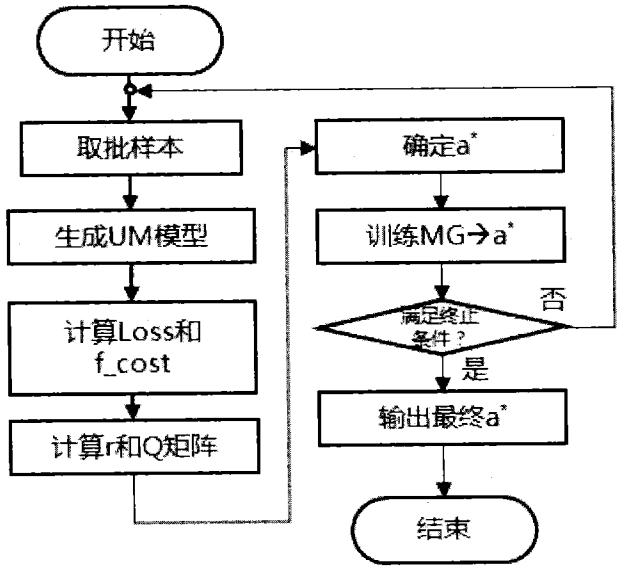
开始
取批样本
生成UM模型
计算Loss和
f_cost
计算r和Q矩阵
确定a*
训练MG→a*
满足终止
条件？
否
是
输出最终a*
结束

图 3

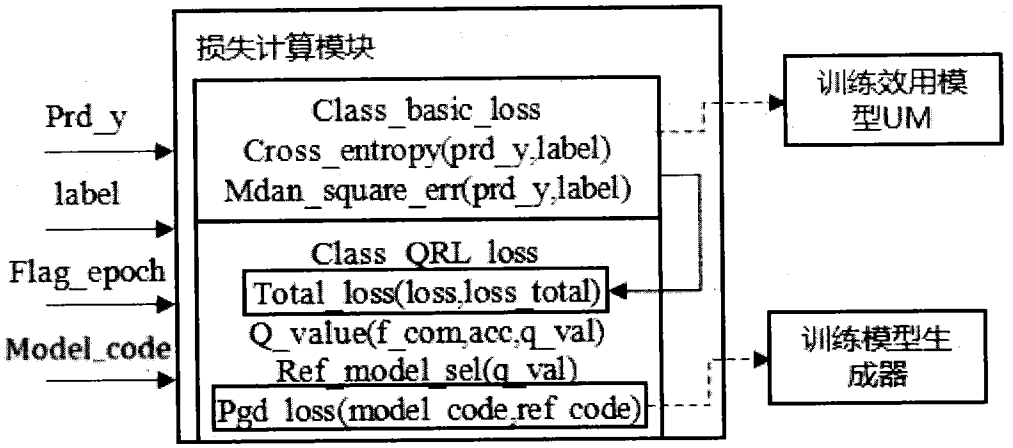
损失计算模块
Prd_y
label
Flag_epoch
Model_code
Class_basic_loss
Cross_entropy(prd_y,label)
Mdan_square_err(prd_y,label)
Class_QRL_loss
Total_loss(loss,loss_total)
Q_value(f_com,acc,q_val)
Ref_model_sel(q_val)
Pgd_loss(model_code,ref_code)
训练效用模
型UM
训练模型生
成器

图 4

参考UM

当前UM	f0	f1	f2	f3	f4
f0	f^0_{cost}	2f1-f0			
f1	2f0-f1	f^1_{cost}			
f2			f^2_{cost}		
f3				f^3_{cost}	
f4					f^4_{cost}

图 5

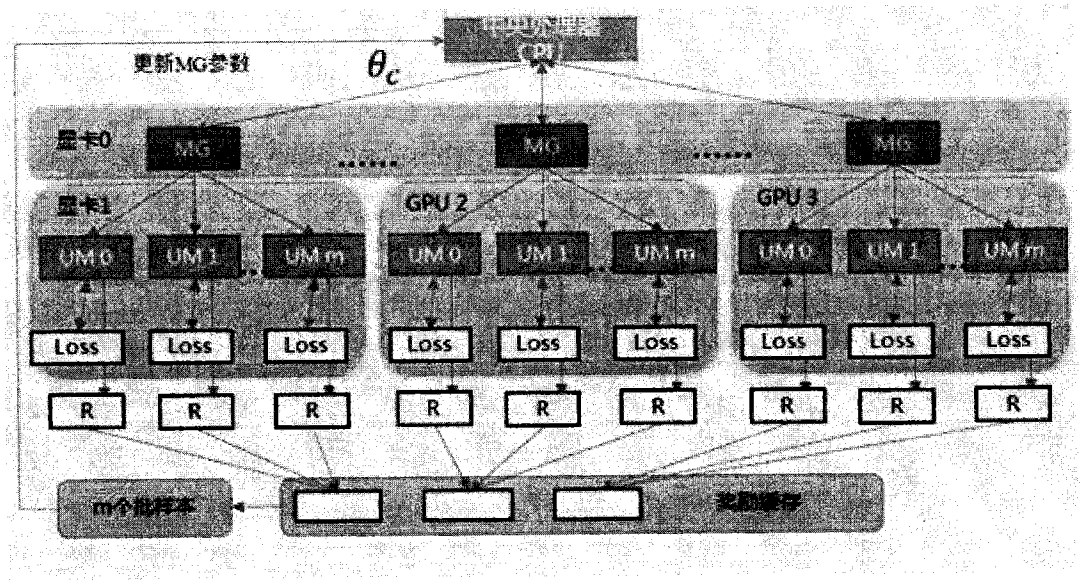

图 6

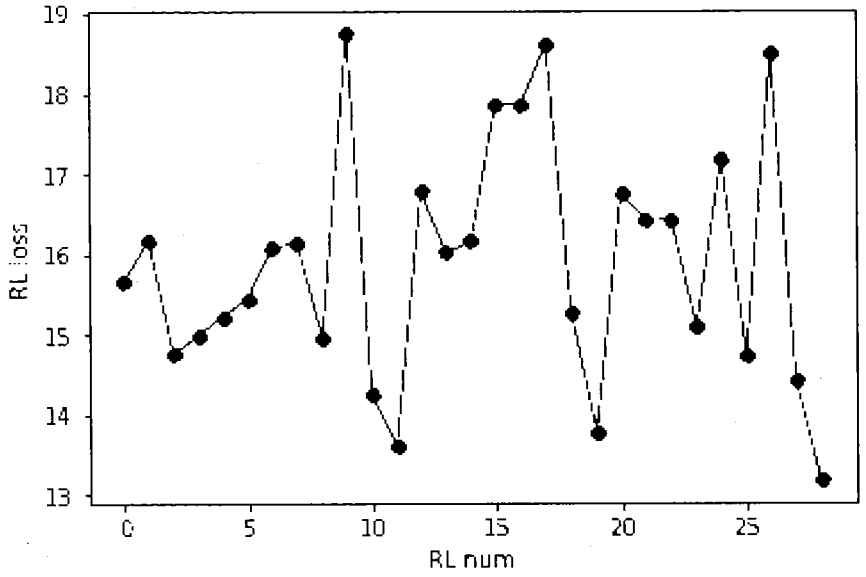
RL loss
RL num
19
18
17
16
15
14
13
0
5
10
15
20
25

图 7

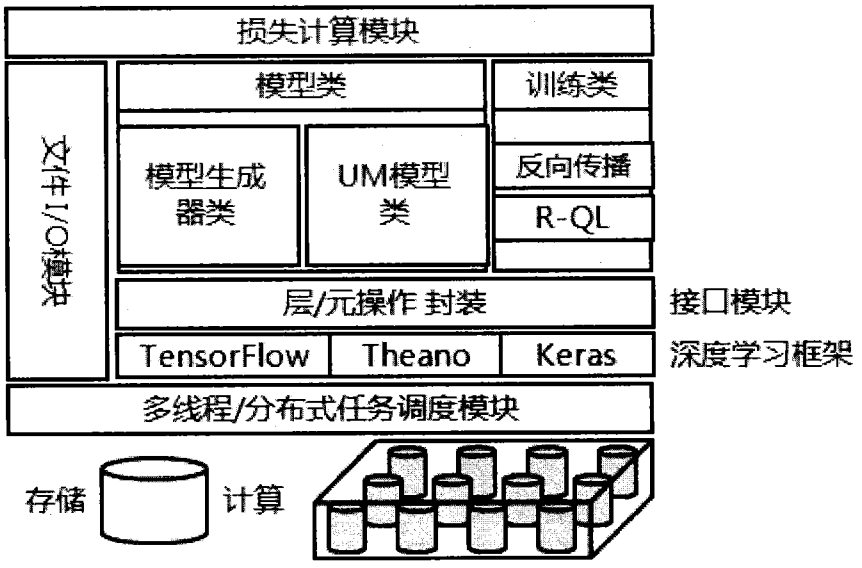
损失计算模块
文件I/O模块
模型类
训练类
模型生成器类
UM模型类
反向传播
R-QL
层/元操作 封装
接口模块
TensorFlow
Theano
Keras
深度学习框架
多线程/分布式任务调度模块
存储
计算

图 8

INTERNATIONAL SEARCH REPORT

International application No.
PCT/CN2019/108984

A. CLASSIFICATION OF SUBJECT MATTER

G06N 20/00(2019.01)i; G06N 3/08(2006.01)i

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

G06N

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

EPODOC, WPI, CNPAT, IEEE, CNKI: 机器学习, 多组, 数据, 网络参数, 模型生成器, 模型编码, 解码器, AutoML, auto machine learning, Batch, Model Generator, model code, Model Decoder, utility model

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
PX	CN 110222847 A (SUZHOU INSPUR INTELLIGENT TECHNOLOGY CO., LTD.) 10 September 2019 (2019-09-10) claims 1-10	1-10
X	CN 109472366 A (ZHENGZHOU YUNHAI INFORMATION TECHNOLOGY CO., LTD.) 15 March 2019 (2019-03-15) description, paragraphs [0076]-[0094], and figures 1, 5, and 6	1-10
A	CN 109816116 A (TENCENT TECHNOLOGY SHENZHEN CO., LTD.) 28 May 2019 (2019-05-28) entire document	1-10

☐ Further documents are listed in the continuation of Box C. ☑ See patent family annex.

* Special categories of cited documents:
"A" document defining the general state of the art which is not considered to be of particular relevance
"E" earlier application or patent but published on or after the international filing date
"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)
"O" document referring to an oral disclosure, use, exhibition or other means
"P" document published prior to the international filing date but later than the priority date claimed
"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art
"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
01 March 2020	**16 March 2020**

Name and mailing address of the ISA/CN	Authorized officer
China National Intellectual Property Administration (ISA/CN) **No. 6, Xitucheng Road, Jimenqiao Haidian District, Beijing 100088** **China**	
Facsimile No. **(86-10)62019451**	Telephone No.

INTERNATIONAL SEARCH REPORT
Information on patent family members

International application No.
PCT/CN2019/108984

Patent document cited in search report	Publication date (day/month/year)	Patent family member(s)	Publication date (day/month/year)
CN 110222847 A	10 September 2019	None	
CN 109472366 A	15 March 2019	None	
CN 109816116 A	28 May 2019	None	

国际检索报告

国际申请号
PCT/CN2019/108984

A. 主题的分类

G06N 20/00(2019.01)i; G06N 3/08(2006.01)i

按照国际专利分类(IPC)或者同时按照国家分类和IPC两种分类

B. 检索领域

检索的最低限度文献(标明分类系统和分类号)

G06N

包含在检索领域中的除最低限度文献以外的检索文献

在国际检索时查阅的电子数据库(数据库的名称，和使用的检索词（如使用））

EPODOC, WPI, CNPAT, IEEE, CNKI：机器学习，多组，数据，网络参数，模型生成器，模型编码，解码器，AutoML, auto machine learning, Batch, Model Generator, model code, Model Decoder, utility model

C. 相关文件

类 型*	引用文件，必要时，指明相关段落	相关的权利要求
PX	CN 110222847 A (苏州浪潮智能科技有限公司) 2019年 9月 10日 (2019-09-10) 权利要求1-10	1-10
X	CN 109472366 A (郑州云海信息技术有限公司) 2019年 3月 15日 (2019-03-15) 说明书第[0076]-[0094]段，附图1，5-6	1-10
A	CN 109816116 A (腾讯科技深圳有限公司) 2019年 5月 28日 (2019-05-28) 全文	1-10

☐ 其余文件在C栏的续页中列出。

☑ 见同族专利附件。

* 引用文件的具体类型：

“A” 认为不特别相关的表示了现有技术一般状态的文件

“E” 在国际申请日的当天或之后公布的在先申请或专利

“L” 可能对优先权要求构成怀疑的文件，或为确定另一篇引用文件的公布日而引用的或者因其他特殊理由而引用的文件（如具体说明的）

“O” 涉及口头公开、使用、展览或其他方式公开的文件

“P” 公布日先于国际申请日但迟于所要求的优先权日的文件

“T” 在申请日或优先权日之后公布，与申请不相抵触，但为了理解发明之理论或原理的在后文件

“X” 特别相关的文件，单独考虑该文件，认定要求保护的发明不是新颖的或不具有创造性

“Y” 特别相关的文件，当该文件与另一篇或者多篇该类文件结合并且这种结合对于本领域技术人员为显而易见时，要求保护的发明不具有创造性

“&” 同族专利的文件

国际检索实际完成的日期	国际检索报告邮寄日期
2020年 3月 1日	2020年 3月 16日
ISA/CN的名称和邮寄地址 中国国家知识产权局(ISA/CN) 中国北京市海淀区蓟门桥西土城路6号 100088 传真号 (86-10)62019451	受权官员 李铎 电话号码 86-(10)-53961393

国际检索报告
关于同族专利的信息

国际申请号
PCT/CN2019/108984

检索报告引用的专利文件	公布日 (年/月/日)	同族专利	公布日 (年/月/日)
CN 110222847 A	2019年 9月 10日	无	
CN 109472366 A	2019年 3月 15日	无	
CN 109816116 A	2019年 5月 28日	无	