



(21) 申请号 201911352575.9

(22) 申请日 2019.12.25

(65) 同一申请的已公布的文献号
申请公布号 CN 113033800 A

(43) 申请公布日 2021.06.25

(73) 专利权人 香港理工大学深圳研究院
地址 518057 广东省深圳市南山区高新园
南区粤兴一道18号香港理工大学产学研
研大楼205室

(72) 发明人 郭嵩 周祺华 詹玉峰

(74) 专利代理机构 深圳中一专利商标事务所
44237
专利代理师 徐启艳

(51) Int. Cl.
G06N 3/08 (2023.01)
G06N 3/0464 (2023.01)

(56) 对比文件

CN 109558909 A, 2019.04.02

CN 107018184 A, 2017.08.04

CN 109299781 A, 2019.02.01

CN 109635922 A, 2019.04.16

CN 109710289 A, 2019.05.03

CN 109871995 A, 2019.06.11

US 2018060759 A1, 2018.03.01

Suyog Gupta et al.. “Model Accuracy
and Runtime Tradeoff in Distributed Deep
Learning: A Systematic Study”.《IEEE》
.2017,第171-180页.

Suyog Gupta et al.. “Model Accuracy
and Runtime Tradeoff in Distributed Deep
Learning: A Systematic Study”.《IEEE》
.2017,第171-180页.

审查员 赵慧敏

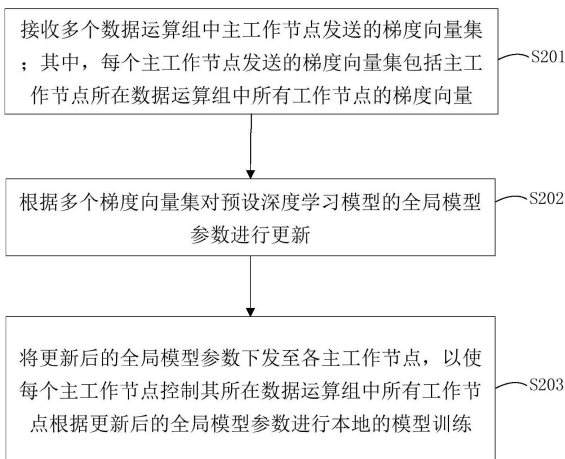
权利要求书2页 说明书14页 附图7页

(54) 发明名称

分布式深度学习方法、装置、参数服务器及
主工作节点

(57) 摘要

本申请属于计算机技术领域,提供了一种分
布式深度学习方法、装置、参数服务器及主工作
节点。方法包括接收多个数据运算组中主工作节
点发送的梯度向量集;其中,每个主工作节点发
送的梯度向量集包括该主工作节点所在数据运
算组中所有工作节点的梯度向量;根据多个梯度
向量集对预设深度学习模型的全局模型参数进
行更新;将更新后的全局模型参数下发至各主工
作节点,以使每个主工作节点控制其所在数据运
算组中所有工作节点根据更新后的全局模型参
数进行本地的模型训练。本申请实施例的分布式
深度学习方法以数据运算组为粒度进行数据以
及任务的调度,减少了每次迭代中与参数服务器
同步的数据量,降低通信开销且提高了各工作节
点的资源利用率。



1. 一种分布式深度学习方法,其特征在于,包括:

接收多个数据运算组中主工作节点发送的梯度向量集;其中,每个主工作节点发送的梯度向量集包括所述主工作节点所在数据运算组中所有工作节点的梯度向量,所述主工作节点为参数服务器指定的数据运算中带宽最大的工作节点或者计算能力最大的工作节点;

根据多个所述梯度向量集对预设深度学习模型的全局模型参数进行更新;

将更新后的全局模型参数下发至各主工作节点,以使所述每个主工作节点控制其所在数据运算组中所有工作节点根据所述更新后的全局模型参数进行本地的模型训练。

2. 如权利要求1所述的分布式深度学习方法,其特征在于,所述方法还包括:

获取多个工作节点的运行状态信息;

根据所述多个运行状态信息,对所述多个工作节点进行重新分组,获得多个更新后的数据运算组;

确定每个更新后的数据运算组的主工作节点。

3. 如权利要求2所述的分布式深度学习方法,其特征在于,所述运行状态信息包括计算能力信息、通信能力信息以及贡献能力信息;其中,所述计算能力信息用于描述每个工作节点计算能力的大小;所述通信能力信息用于描述每个工作节点的可用带宽;所述贡献能力信息用于描述每个工作节点对所述深度学习模型训练过程的贡献度大小。

4. 如权利要求3所述的分布式深度学习方法,其特征在于,所述根据所述多个运行状态信息,对所述多个工作节点进行重新分组,获得多个更新后的数据运算组,包括:

根据所述多个运行状态信息生成评价指标,所述评价指标包括所述计算能力信息的权重值、所述通信能力信息的权重值以及所述贡献能力信息的权重值;

根据所述评价指标计算获得所述多个工作节点中每两个工作节点之间的距离;其中,所述距离用于描述两个工作节点之间的相似度;

根据所述多个工作节点中每两个工作节点之间的距离,对所述多个工作节点进行重新分组,获得多个更新后的数据运算组。

5. 如权利要求1所述的分布式深度学习方法,其特征在于,所述接收多个数据运算组中主工作节点发送的梯度向量集,包括:

针对每个数据运算组,在接收到所述数据运算组中主工作节点发送的梯度向量集时,将该数据运算组的迭代次数加一;

获取所述多个数据运算组中的最大迭代次数和最小迭代次数;

若所述最大迭代次数与所述最小迭代次数之间的差值大于预设阈值,则停止接收所述最大迭代次数对应的数据运算组中主工作节点发送的梯度向量集,直至所述最大迭代次数与最小迭代次数之间的差值小于预设阈值。

6. 一种分布式深度学习方法,其特征在于,包括:

主工作节点获取所述主工作节点所在数据运算组中所有工作节点的梯度向量,并根据多个所述梯度向量生成梯度向量集,所述主工作节点为参数服务器指定的数据运算中带宽最大的工作节点或者计算能力最大的工作节点;

所述主工作节点将所述梯度向量集发送至参数服务器,以使所述参数服务器根据包含所述主工作节点在内的多个主工作节点发送的梯度向量集对预设深度学习模型的全局模型参数进行更新;

接收所述参数服务器返回的更新后的全局模型参数,根据所述更新后的全局模型参数进行本地的模型训练;

将所述更新后的全局模型参数下发至本工作节点所在数据运算组中的各子工作节点,以使各子工作节点根据所述更新后的全局模型参数进行本地的模型训练。

7.一种分布式深度学习装置,其特征在于,包括:

第一接收模块,用于接收多个数据运算组中主工作节点发送的梯度向量集;其中,每个主工作节点发送的梯度向量集包括所述主工作节点所在数据运算组中所有工作节点的梯度向量,所述主工作节点为参数服务器指定的数据运算中带宽最大的工作节点或者计算能力最大的工作节点;

更新模块,用于根据多个所述梯度向量集对预设深度学习模型的全局模型参数进行更新;

第一下发模块,用于将更新后的全局模型参数下发至各主工作节点,以使所述每个主工作节点控制其所在数据运算组中所有工作节点根据所述更新后的全局模型参数进行本地的模型训练。

8.一种分布式深度学习装置,其特征在于,应用主工作节点,分布式深度学习装置包括:

获取模块,用于获取本工作节点所在数据运算组中所有工作节点的梯度向量,并根据多个所述梯度向量生成梯度向量集,所述主工作节点为参数服务器指定的数据运算中带宽最大的工作节点或者计算能力最大的工作节点;

发送模块,用于将所述梯度向量集发送至参数服务器,以使所述参数服务器根据多个主工作节点发送的梯度向量集对预设深度学习模型的全局模型参数进行更新;

第二接收模块,用于接收所述参数服务器返回的更新后的全局模型参数,根据所述更新后的全局模型参数进行本地的模型训练;

第二下发模块,用于将所述更新后的全局模型参数下发至本工作节点所在数据运算组中的各子工作节点,以使各子工作节点根据所述更新后的全局模型参数进行本地的模型训练。

9.一种参数服务器,包括存储器、处理器以及存储在所述存储器中并可在所述处理器上运行的计算机程序,其特征在于,所述处理器执行所述计算机程序时实现如权利要求1至5任一项所述方法的步骤。

10.一种主工作节点,包括存储器、处理器以及存储在所述存储器中并可在所述处理器上运行的计算机程序,其特征在于,所述处理器执行所述计算机程序时实现如权利要求6所述方法的步骤。

11.一种计算机可读存储介质,所述计算机可读存储介质存储有计算机程序,其特征在于,所述计算机程序被处理器执行时实现如权利要求1至5任一项所述方法的步骤,或实现如权利要求6所述方法的步骤。

分布式深度学习方法、装置、参数服务器及主工作节点

技术领域

[0001] 本申请属于计算机技术领域,尤其涉及一种分布式深度学习方法、装置、参数服务器及主工作节点。

背景技术

[0002] 分布式深度学习是一种解决复杂模型训练以及大数据训练的有效手段。

[0003] 分布式深度学习普遍使用的策略是通过参数服务器将训练模型参数分发到不同的工作节点上,然后这些工作节点并行地使用该模型参数进行模型更新,并将计算结果周期性的同步至参数服务器,直至全局模型达到收敛。

[0004] 由于各工作节点的计算能力、网络带宽差异较大,参数服务器接收到不同工作节点的计算结果的时间差异较大。为了实现数据同步,在每一次迭代中,运算速度较快的工作节点均需要等待运算速度较慢的工作节点,导致大量的工作节点资源利用率低。

发明内容

[0005] 有鉴于此,本申请实施例提供了一种分布式深度学习方法、装置、设备及存储介质,以解决现有技术中分布式深度学习方法中工作节点的资源利用率低的技术问题。

[0006] 第一方面,本申请实施例提供了一种分布式深度学习方法,包括:

[0007] 接收多个数据运算组中主工作节点发送的梯度向量集;其中,每个主工作节点发送的梯度向量集包括主工作节点所在数据运算组中所有工作节点的梯度向量;

[0008] 根据多个梯度向量集对预设深度学习模型的全局模型参数进行更新;

[0009] 将更新后的全局模型参数下发至各主工作节点,以使每个主工作节点控制其所在数据运算组中所有工作节点根据所述更新后的全局模型参数进行本地的模型训练。

[0010] 在第一方面的一种可能的实现方式中,分布式深度学习方法还包括:

[0011] 获取多个工作节点的运行状态信息;

[0012] 根据多个运行状态信息,对多个工作节点进行重新分组,获得多个更新后的数据运算组;

[0013] 确定每个更新后的数据运算组的主工作节点。

[0014] 在第一方面的一种可能的实现方式中,运行状态信息包括计算能力信息、通信能力信息以及贡献能力信息;其中,计算能力信息用于描述每个工作节点计算能力的大小;通信能力信息用于描述每个工作节点的可用带宽;贡献能力信息用于描述每个工作节点对所述深度学习模型训练过程的贡献度大小。

[0015] 在第一方面的一种可能的实现方式中,根据多个运行状态信息,对多个工作节点进行重新分组,获得多个更新后的数据运算组,包括:

[0016] 根据多个运行状态信息生成评价指标,该评价指标包括计算能力信息的权重值、通信能力信息的权重值以及贡献能力信息的权重值;

[0017] 根据评价指标计算获得多个工作节点中每两个工作节点之间的距离;其中,距离

用于描述两个工作节点之间的相似度；

[0018] 根据多个工作节点中每两个工作节点之间的距离，对多个工作节点进行重新分组，获得多个更新后的数据运算组。

[0019] 在第一方面的一种可能的实现方式中，所述接收多个数据运算组中主工作节点发送的梯度向量集，包括：

[0020] 针对每个数据运算组，在接收到该数据运算组中主工作节点发送的梯度向量集时，将该数据运算组的迭代次数加一；

[0021] 获取多个数据运算组中的最大迭代次数和最小迭代次数；

[0022] 若最大迭代次数与最小迭代次数之间的差值大于预设阈值，则停止接收最大迭代次数对应的数据运算组中主工作节点发送的梯度向量集，直至最大迭代次数与最小迭代次数之间的差值小于预设阈值。

[0023] 第二方面，本申请实施例提供了一种分布式深度学习方法，包括：

[0024] 主工作节点获取该主工作节点所在数据运算组中所有工作节点的梯度向量，并根据多个梯度向量生成梯度向量集；

[0025] 主工作节点将梯度向量集发送至参数服务器，以使参数服务器根据包含该主工作节点在内的多个主工作节点发送的梯度向量集对预设深度学习模型的全局模型参数进行更新；

[0026] 接收参数服务器返回的更新后的全局模型参数，根据更新后的全局模型参数进行本地的模型训练；

[0027] 将更新后的全局模型参数下发至本工作节点所在数据运算组中的各子工作节点，以使各子工作节点根据更新后的全局模型参数进行本地的模型训练。

[0028] 第三方面，本申请实施例提供了一种分布式深度学习装置，包括：

[0029] 第一接收模块，用于接收多个数据运算组中主工作节点发送的梯度向量集；其中，每个主工作节点发送的梯度向量集包括所述主工作节点所在数据运算组中所有工作节点的梯度向量；

[0030] 更新模块，用于根据多个所述梯度向量集对预设深度学习模型的全局模型参数进行更新；

[0031] 第一下发模块，用于将更新后的全局模型参数下发至各主工作节点，以使所述每个主工作节点控制其所在数据运算组中所有工作节点根据所述更新后的全局模型参数进行本地的模型训练。

[0032] 第四方面，本申请实施例提供了一种分布式深度学习装置，应用于主工作节点，分布式深度学习装置包括：

[0033] 获取模块，用于获取该主工作节点所在数据运算组中所有工作节点的梯度向量，并根据多个梯度向量生成梯度向量集；

[0034] 发送模块，用于将梯度向量集发送至参数服务器，以使参数服务器根据包含该主工作节点在内的多个主工作节点发送的梯度向量集对预设深度学习模型的全局模型参数进行更新；

[0035] 第二接收模块，用于接收参数服务器返回的更新后的全局模型参数，根据更新后的全局模型参数进行本地的模型训练；

[0036] 第二下发模块,用于将更新后的全局模型参数下发至本工作节点所在数据运算组中的各子工作节点,以使各子工作节点根据更新后的全局模型参数进行本地的模型训练。

[0037] 第五方面,本申请实施例提供了一种参数服务器,包括存储器、处理器以及存储在存储器中并可在处理器上运行的计算机程序,处理器执行计算机程序时实现上述第一方面任一项方法的步骤。

[0038] 第六方面,本申请实施例提供了一种主工作节点,包括存储器、处理器以及存储在存储器中并可在处理器上运行的计算机程序,处理器执行计算机程序时实现上述第二方面任一项方法的步骤。

[0039] 第七方面,本申请实施例提供了一种计算机可读存储介质,计算机可读存储介质存储有计算机程序,计算机程序被处理器执行时实现上述第一方面或第二方面任一项方法的步骤。

[0040] 本申请实施例提供的分布式深度学习方法,参数服务器接收多个数据运算组中主工作节点发送的梯度向量集,根据多个梯度向量集对预设深度学习模型的全局模型参数进行更新,将更新后的全局模型参数下发至各主工作节点,以使每个主工作节点控制其所在数据运算组中所有工作节点根据所述更新后的全局模型参数进行本地的模型训练。本申请实施例提供的分布式深度学习方法,以数据运算组为粒度进行任务调度与资源分配,每个数据运算组的主工作节点发送的梯度向量集包括该数据运算组中所有工作节点的梯度向量,与现有技术中集群中所有工作节点均与参数服务器直接通信相比,极大地减少了运算速度较快的工作节点等待运算速度较慢工作节点的发生频次,进而提高了各工作节点的资源利用率,提高了分布式深度学习系统的收敛速度。

[0041] 另一方面,主工作节点首先获取其所在数据运算组中的所有工作节点的梯度向量,生成梯度向量集后再发送至参数服务器,与现有技术中集群中所有工作节点均与参数服务器直接通信相比,极大地节约了网络开销。

[0042] 可以理解的是,上述第二方面至第七方面的有益效果可以参见上述第一方面中的相关描述,在此不再赘述。

附图说明

[0043] 为了更清楚地说明本申请实施例中的技术方案,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申请的一些实施例,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附图。

[0044] 图1是本申请实施例提供的分布式深度学习系统的架构示意图;

[0045] 图2是本申请一实施例提供的分布式深度学习方法的流程示意图;

[0046] 图3是本申请一实施例提供的将获取多个数据运算组的流程示意图;

[0047] 图4是图3所示实施例中对数据运算组划分的流程示意图;

[0048] 图5是本申请一实施例提供接收多个数据运算组中主工作节点发送的梯度向量集的流程示意图;

[0049] 图6是本申请另一实施例提供的分布式深度学习方法的流程示意图;

[0050] 图7是本申请一实施例提供的分布式深度学习方法的交互信令图;

- [0051] 图8是本申请一实施例提供的分布式深度学习装置的功能框图；
- [0052] 图9是本申请另一实施例提供的分布式深度学习装置的功能框图；
- [0053] 图10是本申请一实施例提供的参数服务器的结构示意图；
- [0054] 图11是本申请一实施例提供的主工作节点的结构示意图。

具体实施方式

[0055] 以下描述中,为了说明而不是为了限定,提出了诸如特定系统结构、技术之类的具体细节,以便透彻理解本申请实施例。然而,本领域的技术人员应当清楚,在没有这些具体细节的其它实施例中也可以实现本申请。在其它情况中,省略对众所周知的系统、装置、电路以及方法的详细说明,以免不必要的细节妨碍本申请的描述。

[0056] 在本申请说明书中描述的参考“一个实施例”或“一些实施例”等意味着在本申请的一个或多个实施例中包括结合该实施例描述的特定特征、结构或特点。由此,在本说明书中的不同之处出现的语句“在一个实施例中”、“在一些实施例中”、“在其他一些实施例中”、“在另外一些实施例中”等不是必然都参考相同的实施例,而是意味着“一个或多个但不是所有的实施例”,除非是以其他方式另外特别强调。术语“包括”、“包含”、“具有”及它们的变形都意味着“包括但不限于”,除非是以其他方式另外特别强调。

[0057] 图1为本申请一实施例提供的分布式深度学习系统的架构示意图。该分布式深度学习系统可适用于大规模异构集群。

[0058] 如图1所示,分布式深度学习系统包括参数服务器10和多个工作节点,参数服务器20对多个工作节点预先进行划分得到多个数据运算组20,每个数据运算组20包括一个主工作节点201和多个子工作节点202。

[0059] 其中,每个数据运算组中的子工作节点个数可以不同。

[0060] 参数服务器10主要用于进行数据运算组的分组决策、任务调度以及资源分配。其通过网络与各数据运算组的主工作节点201进行通信以实现各数据运算组之间的数据同步,每个数据运算组的主工作节点201通过网络与该数据运算组的多个子工作节点202通信,以实现每个数据运算组内的数据同步。

[0061] 为便于理解,图1中参数服务器10与各主工作节点之间的网络连接示例为有线直连方式,但本申请实施例不限于此,网络还可以为无线连接或者间接连接,在此不做限定;针对每个数据运算组,该数据运算组的主工作节点与各子工作节点之间的网络连接同上。

[0062] 本实施例中,同一个数据运算组内的工作节点在主工作节点201的控制下进行数据迭代运算,每一个数据运算组20中的主工作节点201和从工作节点202都需要执行迭代训练任务。主工作节点201可以与参数服务器10直接进行数据传输,每个数据运算组内产生的数据必须通过主工作节点201才能向参数服务器10汇报,即子工作节点202无法直接与参数服务器10通信,必须经过主工作节点201汇聚和转发。

[0063] 本实施例中,分布式深度学习系统还可以包括全局数据存储器,用于存储全局数据集;各工作节点可以从全局数据存储器读取本工作节点需要处理的子数据集。例如:全局数据存储器预先进行分布式虚拟化,以使分布式深度学习系统中的所有工作节点均可以访问该全局数据存储器中的数据。其中,整个全局数据集会被分割成不同的子数据集,分配给不同的工作节点,每个工作节点处理的子数据集预先设定。

[0064] 本申请实施例提供的分布式深度学习系统以数据运算组为粒度进行任务调度与资源分配,每个数据运算组的主工作节点发送的梯度向量集包括该数据运算组中所有工作节点的梯度向量,与现有技术中集群中所有工作节点均与参数服务器直接通信相比,极大地减少了运算速度较快的工作节点等待运算速度较慢工作节点的发生频次,进而提高了各工作节点的资源利用率,提高了分布式深度学习系统的收敛速度。

[0065] 下面以具体地实施例对本申请的技术方案以及本申请的技术方案如何解决上述技术问题进行示例性说明。值得说明的是,下文中列举的具体的实施例可以相互结合,对于相同或相似的概念或过程可能在某些实施例中不再赘述。

[0066] 图2为本申请一实施例提供的分布式深度学习方法的流程示意图,适用于图1所示的分布式深度学习系统,本实施例的执行主体为图1中参数服务器,如图1所示,该分布式深度学习学习方法包括:

[0067] S201、接收多个数据运算组中主工作节点发送的梯度向量集;其中,每个主工作节点发送的梯度向量集包括主工作节点所在数据运算组中所有工作节点的梯度向量。

[0068] 本实施例中,多个数据运算组可以为参数服务器对目标集群中的多个工作节点预先划分获得的,每个数据运算组包括一个主工作节点和多个子工作节点。

[0069] 示例性的,参数服务器可以根据目标集群中多个工作节点的硬件特征、网络拓扑、训练数据、模型复杂度等指标,对目标集群的多个工作节点进行划分,获得多个数据运算组,在获得多个数据运算组中,针对每个数据运算组,参数服务器可以指定该数据运算组中各工作节点当前的带宽最大的工作节点为主工作节点。其中,每个数据运算组作为一个运算分支,执行预设数据集的模型训练任务,以实现目标学习任务。其中,目标学习任务可以为图像分类、图像识别等,可以根据具体的应用需求设置,在此不做限定。

[0070] 其中,每个数据运算组中各工作节点的学习模型以及需处理的数据集预先设置,在每一次迭代训练中,各工作节点仅模型参数与梯度向量发生变化;其中,梯度向量由本工作节点根据当前的模型参数和损失函数计算获得,梯度向量用于表征每个工作节点进行深度学习迭代运算中,使得损失函数下降最快的方向。

[0071] 本实施例中,主工作节点发送的梯度向量集包括主工作节点所在数据运算组中所有工作节点的梯度向量。

[0072] 示例性的,针对每个数据运算组,该数据运算组的各工作节点根据当前的模型参数完成一次迭代训练后,计算生成每个工作节点当前的梯度向量。各子工作节点并将当前的本节点梯度向量发送至该数据运算组的主工作节点;上述主工作节点在接收到本数据运算组中所有子工作节点的梯度向量后,结合本主工作节点的梯度向量生成本数据运算组的梯度向量集,并将该梯度向量集发送至参数服务器。

[0073] 可选地,参数服务器与每个数据运算组的主工作节点之间的数据同步模型可以为延迟控制模式。

[0074] S202、根据多个梯度向量集对预设深度学习模型的全局模型参数进行更新。

[0075] 本实施例中,根据多个梯度向量集对预设深度学习模型的全局模型参数进行更新,包括对多个梯度向量集进行融合处理,生成全局梯度向量,根据全局梯度向量对预设深度学习模型的全局模型参数进行更新。

[0076] 其中,融合处理可以为求平均处理。

[0077] 示例性的,对全局模型参数进行更新处理可以通过下式(1)表示,

$$[0078] \quad w_{t+1} = w_t - \frac{\eta}{P} \sum_{i=1}^P g(w_t^i; \xi_t^i) \quad (1)$$

[0079] 其中, w_t 是全局模型参数, g 为梯度向量, P 为工作节点, ξ_t^i 为每个工作节点待处理的子数据集, η 为系数, ξ_t^i 与 η 均为预设值。

[0080] S203、将更新后的全局模型参数下发至各主工作节点,以使每个主工作节点控制其所在数据运算组中所有工作节点根据更新后的全局模型参数进行本地的模型训练。

[0081] 本实施例中,主工作节点在接收到全局模型参数发送的更新后的全局模型参数后,一方面根据该更新后的全局模型参数进行本地的模型训练;另一方面将该更新后的全局模型参数下发至主工作节点所在数据运算组中的所有子工作节点,以使各子工作节点更具该更新后的全局模型参数进行本地的模型训练。

[0082] 其中,模型训练的过程与现有技术中深度学习模型的模型训练过程相同,在此不再赘述。

[0083] 本申请实施例提供的分布式深度学习方法,参数服务器接收多个数据运算组中主工作节点发送的梯度向量集,根据多个梯度向量集对预设深度学习模型的全局模型参数进行更新,将更新后的全局模型参数下发至各主工作节点,以使每个主工作节点控制其所在数据运算组中所有工作节点根据所述更新后的全局模型参数进行本地的模型训练。本申请实施例提供的分布式深度学习方法,以数据运算组为粒度进行任务调度与资源分配,每个数据运算组的主工作节点发送的梯度向量集包括该数据运算组中所有工作节点的梯度向量,与现有技术中集群中所有工作节点均与参数服务器直接通信相比,极大地减少了运算速度较快的工作节点等待运算速度较慢工作节点的发生频次,进而提高了各工作节点的资源利用率,提高了分布式深度学习系统的收敛速度。

[0084] 另一方面,主工作节点首先获取其所在数据运算组中的所有工作节点的梯度向量,生成梯度向量集后再发送至参数服务器,与现有技术中集群中所有工作节点均与参数服务器直接通信相比,极大地节约了网络开销。

[0085] 图3为本申请一实施例提供的获取多个数据运算组的流程示意图,如图3所示,分布式深度学习方法还包括:

[0086] S301、获取多个工作节点的运行状态信息。

[0087] 本实施例中,多个工作节点可以为目标集群的所有工作节点,也可以仅为目标集群中的部分工作节点。

[0088] 其中,目标集群可以为包括多个工作节点的异构机器集群。

[0089] 在一种实施方式中,分布式深度学习系统在进行训练迭代中,每个工作节点的运行状态会实时发生变化,可以根据各工作节点的运行状态对当前的数据运算组的分组情况进行更新,从而保障数据运算中的各工作节点的具有较高的资源利用率。此时,获取多个工作节点的运行状态信息,可以为参数服务器按照预设的时间间隔接收当前的各主工作节点发送的各主工作节点所在的数据运算组中所有工作节点的当前运行状态信息。

[0090] 在另一种实施方式中,参数服务器首次对各工作节点进行分组时,获取多个工作节点的运行状态信息可以为,参数服务器直接获取所有工作节点的运行状态信息。

[0091] 本实施例中,运行状态信息包括工作节点的计算能力信息、通信能力信息以及贡献能力信息。

[0092] 其中,计算能力信息用于描述每个工作节点计算能力的大小。例如:计算能力信息可以为每轮迭代运算的运算速度。具体地,可以通过每轮迭代运算中计算本节点梯度向量的耗时长短表征各节点的计算能力信息。

[0093] 通信能力信息用于描述每个工作节点当前可用宽带,例如:通信能力信息可以为各子工作节点将预设大小数据发送至对应的主工作节点的耗时长短。

[0094] 贡献能力信息用于描述每个工作节点对深度学习模型训练过程的贡献度大小。示例性的,贡献能力信息可以为本工作节点的累计迭代次数与参数服务器的全局迭代次数的比值。由于不同工作节点的计算能力以及网络带宽不同,导致不同工作节点在进行本地的模型训练时的迭代速率不同,进而导致迭代次数不同。

[0095] S302、根据多个运行状态信息,对多个工作节点进行重新分组,获得多个更新后的数据运算组。

[0096] 本实施例中,参数服务器可以根据多个工作节点的计算能力信息、网络能力信息以及贡献能力信息等指标,对上述多个计算节点进行划分,从而获得多个数据运算组。

[0097] 示例性的,可以将多个工作节点的计算能力信息、网络能力信息以及贡献能力信息作为分类标准,基于聚类算法对多个工作节点进行划分,以使得每个数据运算组中的各工作节点的计算能力、网络带宽等特征相似,而不同数据运算组之间的工作节点的计算能力和网络带宽等特性差异较大。

[0098] S303、确定每个更新后的数据运算组的主工作节点。

[0099] 在获得多个数据运算组中,针对每个数据运算组,参数服务器可以指定该数据运算组中各工作节点当前的带宽最大的工作节点为主工作节点。也可以指定各工作节点中当前的计算能力最大的工作节点作为主工作节点。

[0100] 由于各工作节点的运行状态信息实时变化,本申请实施例提供的分布式深度学习方法中,参数服务器获取多个工作节点的运行状态信息,并根据该运行状态信息对数据运算组进行更新,有效实现对分布式深度学习系统的各工作节点的部署。

[0101] 实际应用中,参数服务器可以按照预设的规则,接收多个工作节点的运行状态信息。例如:预设的规则可以为预设的时间间隔,也可以为深度学习模型的全局模型参数更新预设次数。

[0102] 参数服务器每接收一次多个工作节点的运行状态信息,则对目标集群的运算数据组的划分进行一次更新,有效的实现分布式深度学习系统的弹性部署。且在不同的划分更新中,包含的多个工作节点可以不同。

[0103] 示例性的,假设目标集群包括是S1-S20个计算节点,参数服务器每经过固定的时间获取一次目标集群中多个计算节点的运算状态信息,即对运算数据组的划分进行更新。在第一次分组时,参数服务器接收目标集群中计算节点S1-S15的运行状态信息;在第二次分组时,参数服务器可以接收目标集群中计算节点S1-S16的运行状态信息,或者接收新加入该目标集群中计算节点的运行状态信息,从而实现了分布式深度学习系统的弹性部署。

[0104] 图4为图3所示实施例中数据运算组划分的流程示意图。图4为对步骤302如何获得多个更新后的数据运算组进行了示例性说明。如图4所示,根据多个运行状态信息,对多

个工作节点进行重新分组,获得多个更新后的数据运算组,包括:

[0105] S401、根据多个运行状态信息生成评价指标,该评价指标包括计算能力信息的权重值、通信能力信息的权重值以及贡献能力信息的权重值。

[0106] 本实施例中,通过强化学习(Asynchronous Advantage Actor-Critic(A3C))不断优化产生最适应当前任务状态的内联性评价指标。示例性的,该内联性指标反映为计算能力信息的权重值、通信能力信息的权重值以及贡献能力信息的权重值;具体的,每种信息的权重均为范围0-1的浮点数。

[0107] 示例性的,在分布式深度学习系统开始模型训练的初期,各工作节点的贡献能力信息值相同,各工作节点的通信能力信息是限制深度学习模型迭代收敛速度的关键因素,且差别较大,故此时通信能力信息的权重值大于贡献能力信息的权重值。在分布式深度学习系统进行模型训练的中后期,各工作节点的计算能力成为限制深度学习模型收敛速度的关键因此,此时若进行数据运算组的重新划分,则计算能力信息的权重值最大。

[0108] S402、根据评价指标计算获得多个工作节点中每两个工作节点之间的距离;其中,距离用于描述两个工作节点之间的相似度。

[0109] 本实施例中,可以将评价指标作为输入,基于无监督聚类算法计算获得多个计算节点中各计算节点之间的距离。

[0110] 无监督聚类算法的性能度量可以包括两类,一类为将聚类的结果与参考模型进行比较,获得外部指标,另一类是直接对聚类结果进行评价而不利用任何参考模型,称为内部指标。

[0111] 本实施例中,无监督聚类算法采用内部指标进行聚类的性能度量。例如:该内部指标可以为两个工作节点之间的距离。

[0112] 其中,距离可以为欧氏距离、曼哈顿距离等。

[0113] 本实施例中,在计算获得每两个工作节点之间的距离时,将计算能力信息的权重值、通信能力信息的权重值以及贡献能力信息的权重值作为输入,以使得距离的计算更加符合当前分布式深度学习系统的运行状态。

[0114] S403、根据多个工作节点中每两个工作节点之间的距离,对多个工作节点进行重新分组,获得多个更新后的数据运算组。

[0115] 本实施例中,两个工作节点之间的距离值越大,表示该两个工作节点之间的相似度越小。

[0116] 示例性的,可以为基于密度的聚类算法进行数据运算组的划分,该算法假设聚类结果能够通过样本分布的紧密程度确定,从样本密度的角度来考察样本之间的可连接性,并基于该可连接样本不断扩展聚类簇以获得最终的结果。

[0117] 根据多个计算节点中各计算节点之间的距离里,判断以某一计算节点为核心时,其可密度直达和密度可达到的计算节点,并将由密度可达关系导出的最大密度相连样本集合作为一个数据运算组。

[0118] 本申请实施例提供的分布式深度学习方法,首先基于强化学习方法获取当前运行状态信息中各类信息的权重,并将该权重作为输入,基于聚类算法将多个计算节点划分为多个数据运算组,以使得每个数据运算组中的各工作节点的计算能力和网络带宽等特征相似;每个数据运算组中的各工作节点的计算能力和网络带宽等特征相似,则运行速度最快

的工作节点与运行速度最慢的工作节点在一次迭代中生成梯度向量的耗时相似,故主工作节点进行本数据运算组中各工作节点的梯度向量同步时,等待时间极大的缩短,相同数据运算组中的工作节点的资源利用率均得到了极大的提高。

[0119] 在获得多个数据运算组的分组结果后,参数服务器根据该分组结果对各工作节点的逻辑链接和数据同步模式进行调整。

[0120] 示例性,参数服务器建立参数服务器和各主工作节点之间的通信链路,并将参数服务器与每个数据运算组的主工作节点之间的数据同步模型设置为延迟控制模式。下面通过图5所示的实施例对参数服务基于延迟控制模式接收多个主工作节点发送的梯度向量集的方案进行示例性说明。

[0121] 图5为是本申请一实施例提供接收多个数据运算组中主工作节点发送的梯度向量集的流程示意图。如图5所示,接收多个数据运算组中主工作节点发送的梯度向量集,包括:

[0122] S501、针对每个数据运算组,在接收到该数据运算组中主工作节点发送的梯度向量集时,将该数据运算组的迭代次数加一。

[0123] 数据运算组中的主工作节点每向参数服务器发送一次梯度向量集,则表示该数据运算组完成一次迭代运算,则参数服务器将该数据运算组的迭代次数加一。

[0124] S502、获取多个数据运算组中的最大迭代次数和最小迭代次数。

[0125] 由于不同数据运算组之间的计算能力和网络带宽差别较大,故在相同时间内,不同数据运算组的迭代次数不同。

[0126] S503、若最大迭代次数与最小迭代次数之间的差值大于预设阈值,则停止接收最大迭代次数对应的数据运算组中主工作节点发送的梯度向量集,直至最大迭代次数与最小迭代次数之间的差值小于预设阈值。

[0127] 其中,预设阈值为延时控制模式中的延迟迭代次数,用于控制多个数据运算组之间的延迟差。

[0128] 示例性的,假设分布式深度学习系统包括一个参数服务器和16个工作节点,16个工作节点被参数服务器预先划分为4个数据运算组A1-A4,每个数据运算组用于进行预设数据子集的数据运算。A1-A4分别包括4个工作节点,例如:数据运算组A1包括四个工作节点a1-a4,a1为主工作节点,a2-a4为子工作节点。

[0129] 四个数据运算组A1-A4的计算能力与网络带宽不同,在相同时间间隔时,A1-A4的迭代次数不同,具体表征为向参数服务器发送的梯度向量集的次数不同。

[0130] 预先设置延时控制模式中的延迟迭代次数为T,参数服务器接收各工作节点发送的梯度向量集,接收到每个数据运算组中主工作节点发送的梯度向量集时,将该数据运算组的迭代次数加一,进而获得A1-A4四个数据运算组的迭代次数,判断最大迭代次数与最小迭代次数之间的差值是否大于预设延迟迭代次数T;若是,则停止接收最大迭代次数对应的数据运算组中主工作节点(例如a1)发送的梯度向量集,仅与其他数据运算组(A2-A4)进行数据的交互,,此时A2-A4数据运算组的迭代次数持续增加,直至最大迭代次数与最小迭代次数之间的差值小于预设延迟迭代次数T,参数服务器开始接收a1发送的梯度向量集。

[0131] 本实施例提供的分布式深度学习方法,基于延迟控制模式实现各主工作节点与参数服务器之间的数据同步,保障了收敛的精度,达到了收敛速度和收敛精度的良好平衡。

[0132] 图6为本发明另一实施例提供的分布式深度学习方法的流程示意图。适用于图1所

示的分布式深度学习系统。图6所示分布式深度学习方法的执行主体为图1中任一主工作节点。

[0133] 如图6所示,分布式深度学习方法包括:

[0134] S601、获取本工作节点所在数据运算组中所有工作节点的梯度向量,并根据多个梯度向量生成梯度向量集。

[0135] S602、将梯度向量集发送至参数服务器,以使参数服务器根据多个主工作节点发送的梯度向量集对预设深度学习模型的全局模型参数进行更新。

[0136] S603、接收参数服务器返回的更新后的全局模型参数,根据更新后的全局模型参数进行本地的模型训练。

[0137] 本实施例中,主工作节点根据更新后的全局模型参数进行本地的模型训练包括:根据更新后的全局模型参数进行训练,获得损失函数的取值。

[0138] 其中,损失函数可以为最小绝对值偏差损失函数、最小平方误差损失函数等,在此不做具体限定。例如:损失函数可以用于计算深度学习网络输出的预测图像与样本图像之间的误差值;可以为图像分辨率差值、图像清晰度差值或图像相似度差值等。

[0139] 判断迭代是否满足预设条件,该预设条件可以为当前的训练次数达到预设迭代次数。或者本次迭代训练获得的预测结果与训练样本中的目标结果之间的误差小于或等于预设误差阈值。

[0140] 若否,则根据更新后的全局模型参数对本子工作节点的梯度向量进行更新。

[0141] 若是,则保存当前的模型参数

[0142] S604、将更新后的全局模型参数下发至本工作节点所在数据运算组中的各子工作节点,以使各子工作节点根据更新后的全局模型参数进行本地的模型训练。

[0143] 各子节点根据更新后的全局模型参数进行本地的模型训练的技术方案与步骤603中进行本地的模型训练的技术方案相同。各子工作节点在迭代未满足预设条件时,根据更新后的全局模型参数对本子工作节点的梯度向量进行更新,并将更新后的梯度向量发送至主工作节点。

[0144] 此时主工作节点返回执行步骤601,进行循环迭代直至每个工作节点均满足预设的迭代条件。

[0145] 本实施例中,主工作节点与该数据运算组中各子工作节点之间的数据同步模式设置为块同步模式。具体表征为主工作节点获取所在的数据运算组中所有工作节点的梯度向量后,再根据多个梯度向量生成梯度向量集,并将该梯度向量集发送至参数服务器;若存在任何一个工作节点的梯度向量无法获取,则等待直至获取到该工作节点的梯度向量。即每个数据运算中内各工作节点与参数服务器的数据同步过程是同时的,不存在延时控制,从而保障全局模型参数的准确性。

[0146] 本实施例中其他步骤方法的具体实现原理和过程可以参见上述任一实施例,此处不再赘述。

[0147] 本申请实施例提供的分布式深度学习方法,主工作节点首先获取其所在数据运算组中的所有工作节点的梯度向量,生成梯度向量集后再发送至参数服务器,以数据运算组为粒度进行任务调度与资源分配,每个数据运算组的主工作节点发送的梯度向量集包括该数据运算组中所有工作节点的梯度向量,与现有技术中集群中所有工作节点均与参数服务

器直接通信相比,极大地减少了运算速度较快的工作节点等待运算速度较慢工作节点的发生频次,进而提高了各工作节点的资源利用率,提高了分布式深度学习系统的收敛速度。

[0148] 图7为本发明一实施例提供的分布式深度学习方法的交互信令图,适用于图1所示的分布式深度学习系统,分布式深度学习系统中各数据运算组的处理步骤相同,在本实施例中仅示出一个数据运算组的数据交互过程。该数据运算组包括一个主工作节点和多个子工作节点。由于多个子工作节点的数据处理过程相同,本实施例中通过两个子工作节点进行示例性说明,两个子工作节点分别为子工作节点1和子工作节点n。

[0149] 如图7所示,分布式深度学习方法可以包括:

[0150] S701、子工作节点1和子工作节点n分别获取本子工作节点的梯度向量。

[0151] S702、子工作节点1和第二工作节点分别将梯度向量发送至第一主工作节点。

[0152] S703、主工作节点获取本主工作节点的梯度向量。

[0153] S704、主工作节点生成本主工作节点所在数据运算组的梯度向量集。

[0154] S705、主工作节点将该数据运算组的梯度向量集发送至参数服务器。

[0155] S706、参数服务器根据多个主工作节点发送的梯度向量集对预设深度学习模型的全局模型参数进行更新。

[0156] S707、参数服务器将更新后的全局模型参数发送至主工作节点。

[0157] S708、第一主工作节点将更新后的全局模型参数分别发送至子工作节点1和子工作节点n。

[0158] S709、主工作节点根据更新后的全局模型参数进行本地的模型训练。

[0159] S710、子工作节点1和子工作节点n分别根据更新后的全局模型参数进行本地的模型训练。

[0160] S711、子工作节点1或子工作节点n判定是否满足预设条件。

[0161] S712、若否,子工作节点1或子工作节点n根据更新后的全局模型参数对本子工作节点的梯度向量进行更新,并返回执行步骤S701。

[0162] S713、若是,子工作节点1或子工作节点n保存当前的模型参数。

[0163] 第一工作节点在步骤709之后的处理与S711-S713相同,在此不再赘述。

[0164] 本实施例的具体实施方式可以参见图2图6所示的实施例,此处不再赘述。

[0165] 应理解,上述实施例中各步骤的序号的大小并不意味着执行顺序的先后,各过程的执行顺序应以其功能和内在逻辑确定,而不对本申请实施例的实施过程构成任何限定。

[0166] 基于上述实施例所提供的分布式深度学习,本发明实施例进一步给出实现上述方法实施例的装置实施例。

[0167] 图8为本发明一实施例提供的分布式深度学习装置的结构示意图。该分布式深度学习装置适用于图1所示的分布式深度学习系统的参数服务器。如图8所示,该分布式深度学习装置80包括接收模块801、更新模块802以及第一下发模块803。

[0168] 接收模块801,用于接收多个数据运算组中主工作节点发送的梯度向量集;其中,每个主工作节点发送的梯度向量集包括主工作节点所在数据运算组中所有工作节点的梯度向量。

[0169] 更新模块802,用于根据多个梯度向量集对预设深度学习模型的全局模型参数进

行更新。

[0170] 第一下发模块803,用于将更新后的全局模型参数下发至各主工作节点,以使每个主工作节点控制其所在数据运算组中所有工作节点根据所述更新后的全局模型参数进行本地的模型训练。

[0171] 接收模块801,具体用于:

[0172] 针对每个数据运算组,在接收到该数据运算组中主工作节点发送的梯度向量集时,将该数据运算组的迭代次数加一;

[0173] 获取多个数据运算组中的最大迭代次数和最小迭代次数;

[0174] 若最大迭代次数与最小迭代次数之间的差值大于预设阈值,则停止接收最大迭代次数对应的数据运算组中主工作节点发送的梯度向量集,直至最大迭代次数与最小迭代次数之间的差值小于预设阈值。

[0175] 可选地,分布式深度学习装置80还包括分组模块。

[0176] 分组模块,用于:

[0177] 获取多个工作节点的运行状态信息;

[0178] 根据多个运行状态信息,对多个工作节点进行重新分组,获得多个更新后的数据运算组;

[0179] 确定每个更新后的数据运算组的主工作节点。

[0180] 可选地,运行状态信息包括计算能力信息、通信能力信息以及贡献能力信息;其中,计算能力信息用于描述每个工作节点计算能力的大小;通信能力信息用于描述每个工作节点的可用带宽;贡献能力信息用于描述每个工作节点对所述深度学习模型训练过程的贡献度大小。

[0181] 分组模块,具体用于:

[0182] 根据多个运行状态信息生成评价指标,该评价指标包括计算能力信息的权重值、通信能力信息的权重值以及贡献能力信息的权重值;

[0183] 根据评价指标计算获得多个工作节点中每两个工作节点之间的距离;其中,距离用于描述两个工作节点之间的相似度;

[0184] 根据多个工作节点中每两个工作节点之间的距离,对多个工作节点进行重新分组,获得多个更新后的数据运算组。

[0185] 图8所示实施例的分布式深度学习装置可用于执行上述方法中以参数服务器为执行主体的技术方案,本实施例此处不再赘述。

[0186] 图9为本发明另一实施例提供的分布式深度学习装置的结构示意图。该分布式深度学习装置适用于图1所示的分布式深度学习系统中任一主工作节点。如图9所示,该分布式深度学习装置包括获取模块901,发送模块902、第二接收模块903以及第二下发模块904;

[0187] 获取模块901,用于获取该主工作节点所在数据运算组中所有工作节点的梯度向量,并根据多个梯度向量生成梯度向量集;

[0188] 发送模块902,用于将梯度向量集发送至参数服务器,以使参数服务器根据包含该主工作节点在内的多个主工作节点发送的梯度向量集对预设深度学习模型的全局模型参数进行更新;

[0189] 第二接收模块903,用于接收参数服务器返回的更新后的全局模型参数,根据更新

后的全局模型参数进行本地的模型训练；

[0190] 第二下发模块904,用于将更新后的全局模型参数下发至本工作节点所在数据运算组中的各子工作节点,以使各子工作节点根据更新后的全局模型参数进行本地的模型训练。

[0191] 图9所示实施例的分布式深度学习装置可用于执行上述方法中以主工作节点为执行主体的技术方案,本实施例此处不再赘述。

[0192] 所属领域的技术人员可以清楚地了解到,为了描述的方便和简洁,仅以上述各功能单元、模块的划分进行举例说明,实际应用中,可以根据需要而将上述功能分配由不同的功能单元、模块完成,即将所述装置的内部结构划分成不同的功能单元或模块,以完成以上描述的全部或者部分功能。实施例中的各功能单元、模块可以集成在一个处理单元中,也可以是各个单元单独物理存在,也可以两个或两个以上单元集成在一个单元中,上述集成的单元既可以采用硬件的形式实现,也可以采用软件功能单元的形式实现。另外,各功能单元、模块的具体名称也只是为了便于相互区分,并不用于限制本申请的保护范围。上述系统中单元、模块的具体工作过程,可以参考前述方法实施例中的对应过程,在此不再赘述。

[0193] 图10是本申请一实施例提供的参数服务器的示意图。如图11所示,该实施例的参数服务器100包括:至少一个处理器1001、存储器1002以及存储在所述存储器1002中并可在所述处理器1001上运行的计算机程序。参数服务器还包括通信部件1003,其中,处理器1001、存储器1002以及通信部件1003通过总线1004连接。处理器1001执行所述计算机程序时实现上述以参数服务器为执行主体的分布式深度学习方法实施例中的步骤。

[0194] 图11是本申请一实施例提供的主工作节点的示意图。如图11所示,该实施例的主工作节点110包括:至少一个处理器1101、存储器1102以及存储在所述存储器1102中并可在所述处理器1101上运行的计算机程序。主工作节点还包括通信部件1103,其中,处理器1101、存储器1102以及通信部件1103通过总线1004连接。各主工作节点110与参数服务器之间可以通过总线1104连接。处理器1101执行所述计算机程序时实现上述以主工作节点为执行主体的分布式深度学习方法实施例中的步骤。

[0195] 图10和图11中所称处理器可以是中央处理单元(Central Processing Unit, CPU),还可以是其他通用处理器、数字信号处理器(Digital Signal Processor, DSP)、专用集成电路(Application Specific Integrated Circuit, ASIC)、现成可编程门阵列(Field-Programmable Gate Array, FPGA)或者其他可编程逻辑器件、分立门或者晶体管逻辑器件、分立硬件组件等。通用处理器可以是微处理器或者该处理器也可以是任何常规的处理器等。

[0196] 总线可以是工业标准体系结构(Industry Standard Architecture, ISA)总线、外部设备互连(Peripheral Component, PCI)总线或扩展工业标准体系结构(Extended Industry Standard Architecture, EISA)总线等。总线可以分为地址总线、数据总线、控制总线等。为便于表示,本申请附图中的总线并不限定仅有一根总线或一种类型的总线。

[0197] 本申请实施例还提供了一种计算机可读存储介质,其上存储有计算机程序,该计算机程序被处理器执行以实现上述以参数服务器为执行主体时的任一实现方式所述的方法,或者实现上述以主工作及节点为执行主体时的任一实现方式所述的方法。

[0198] 上述可读存储介质可以是由任何类型的易失性或非易失性存储设备或者它们的

组合实现,如静态随机存取存储器(SRAM),电可擦除可编程只读存储器(EEPROM),可擦除可编程只读存储器(EPROM),可编程只读存储器(PROM),只读存储器(ROM),磁存储器,快闪存储器,磁盘或光盘。可读存储介质可以是通用或专用计算机能够存取的任何可用介质。

[0199] 在上述实施例中,对各个实施例的描述都各有侧重,某个实施例中未详述或记载的部分,可以参见其它实施例的相关描述。

[0200] 本领域普通技术人员可以意识到,结合本文中所公开的实施例描述的各示例的单元及算法步骤,能够以电子硬件、或者计算机软件和电子硬件的结合来实现。这些功能究竟以硬件还是软件方式来执行,取决于技术方案的特定应用和设计约束条件。专业技术人员可以对每个特定的应用来使用不同方法来实现所描述的功能,但是这种实现不应认为超出本申请的范围。

[0201] 在本申请所提供的实施例中,应该理解到,所揭露的装置/网络设备和/或方法,可以通过其它的方式实现。所述作为分离部件说明的单元可以是或者也可以不是物理上分开的,作为单元显示的部件可以是或者也可以不是物理单元,即可以位于一个地方,或者也可以分布到多个网络单元上。可以根据实际的需要选择其中的部分或者全部单元来实现本实施例方案的目的。

[0202] 以上所述实施例仅用以说明本申请的技术方案,而非对其限制;尽管参照前述实施例对本申请进行了详细的说明,本领域的普通技术人员应当理解:其依然可以对前述各实施例所记载的技术方案进行修改,或者对其中部分技术特征进行等同替换;而这些修改或者替换,并不使相应技术方案的本质脱离本申请各实施例技术方案的精神和范围,均应包含在本申请的保护范围之内。

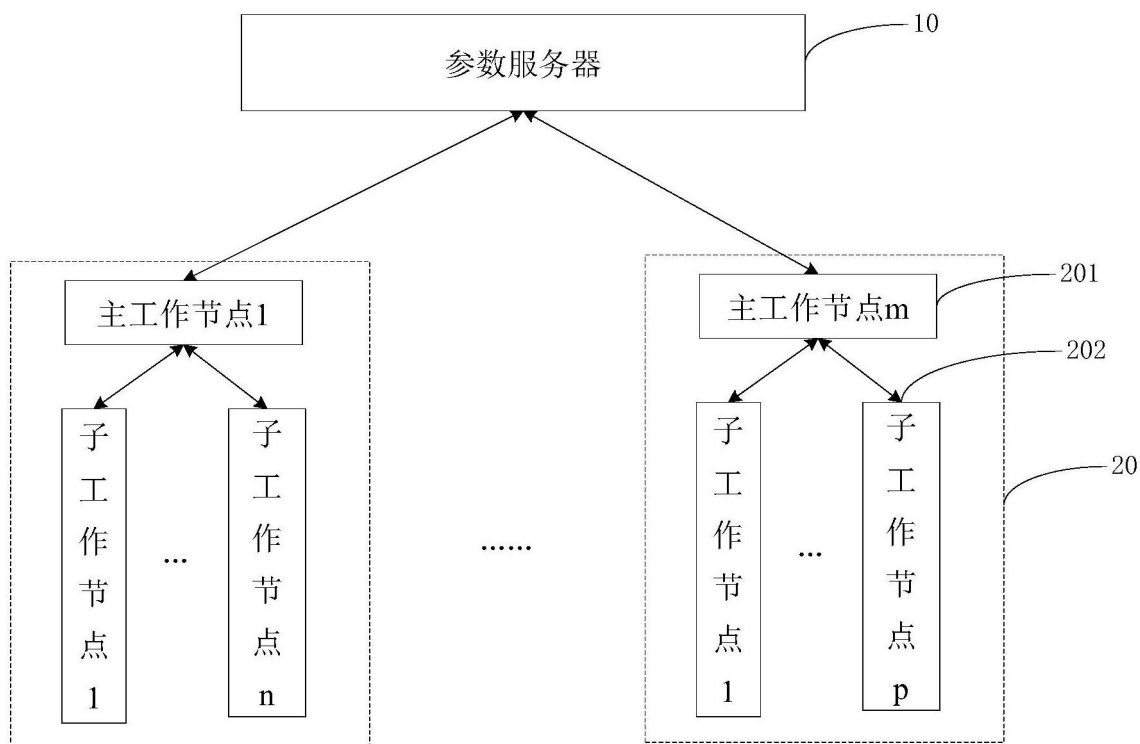


图1

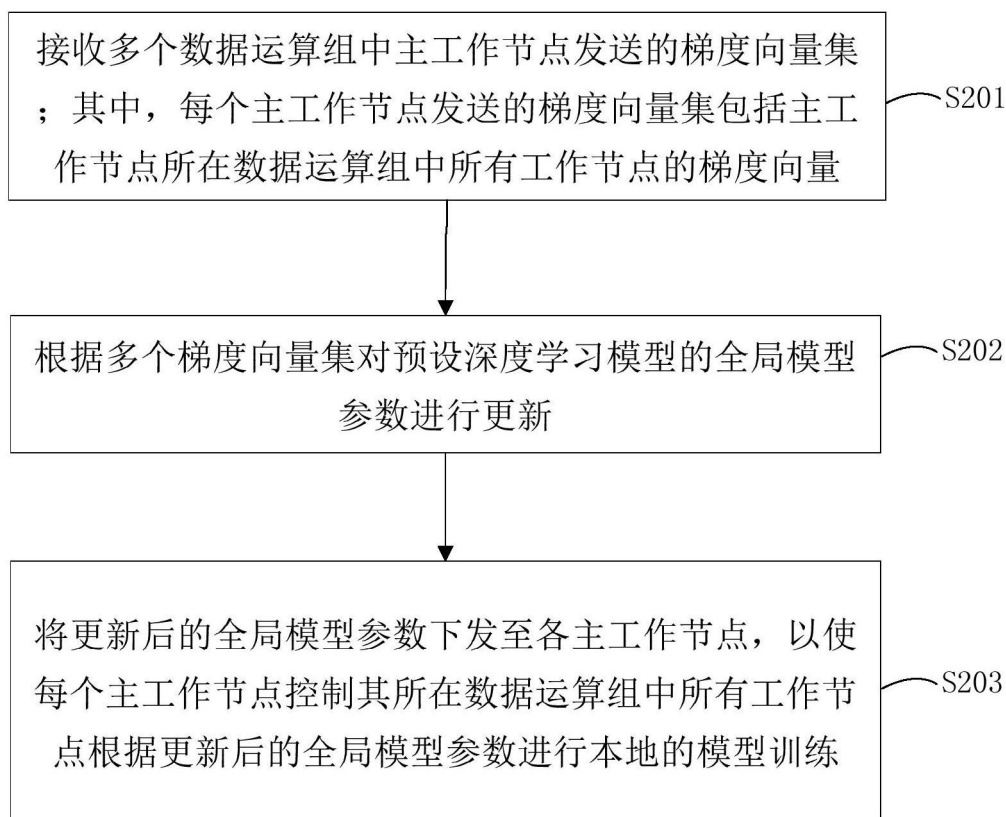


图2

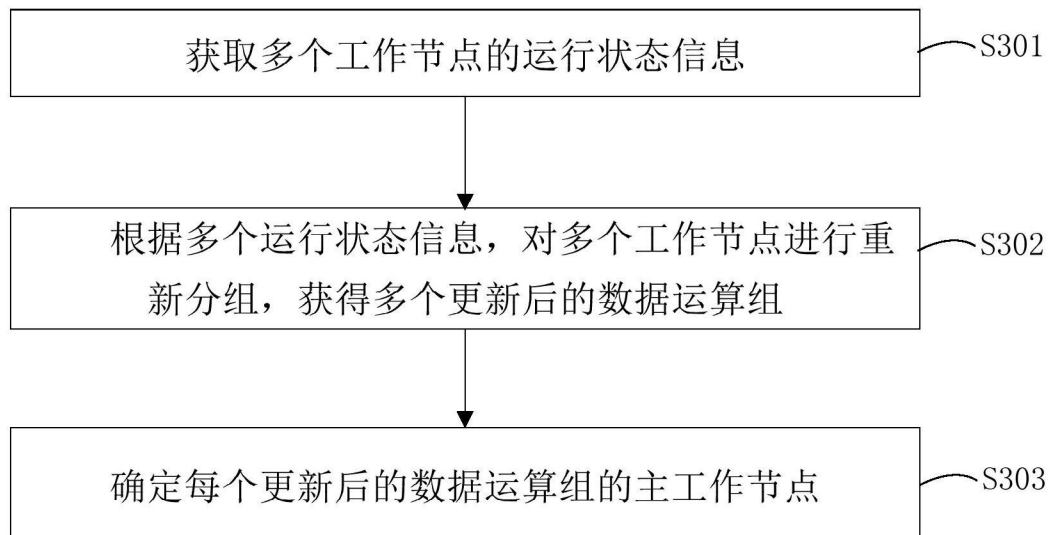


图3

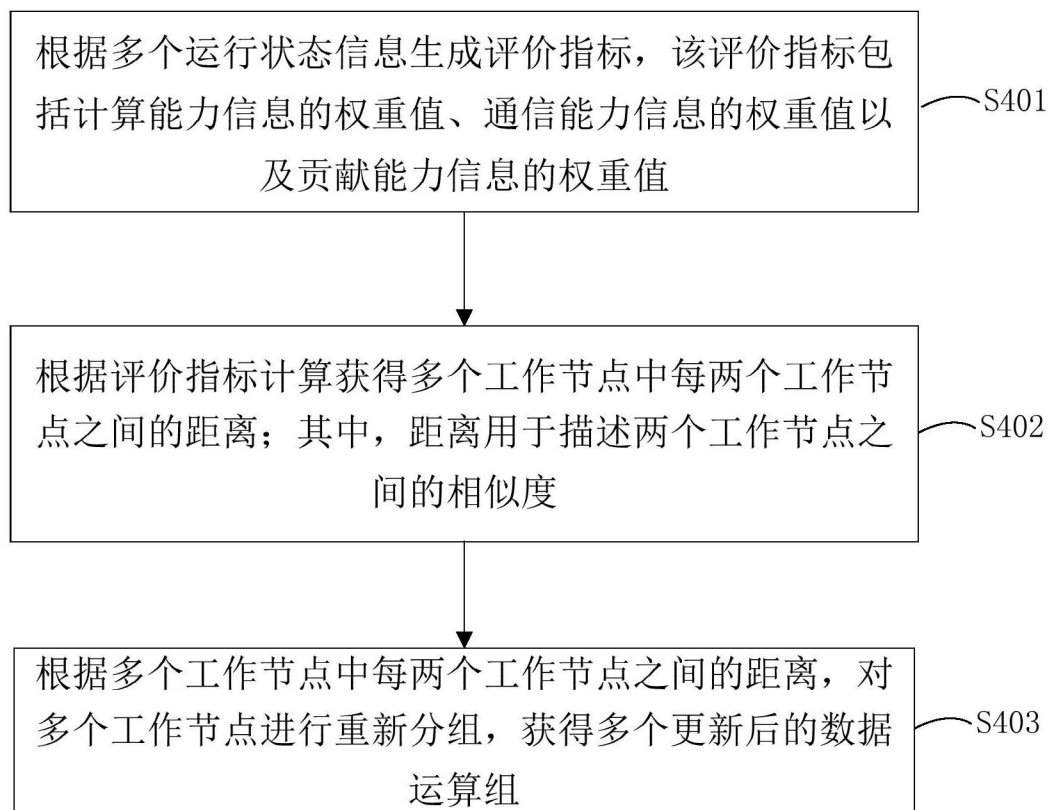


图4

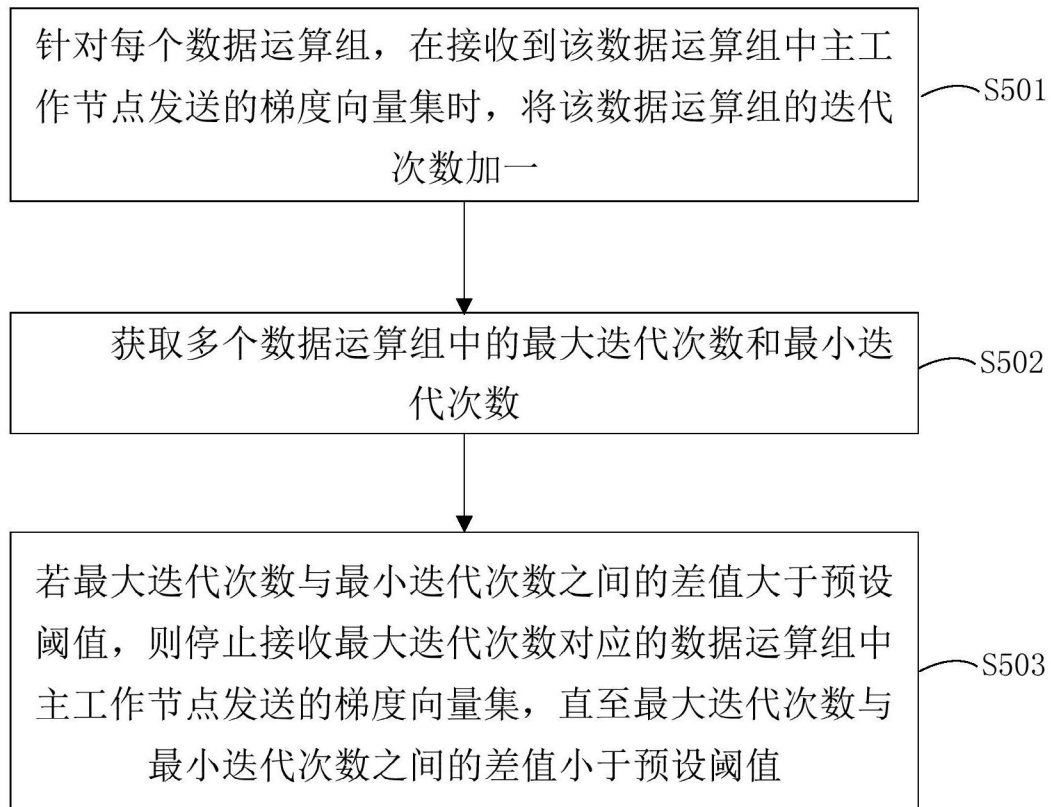


图5

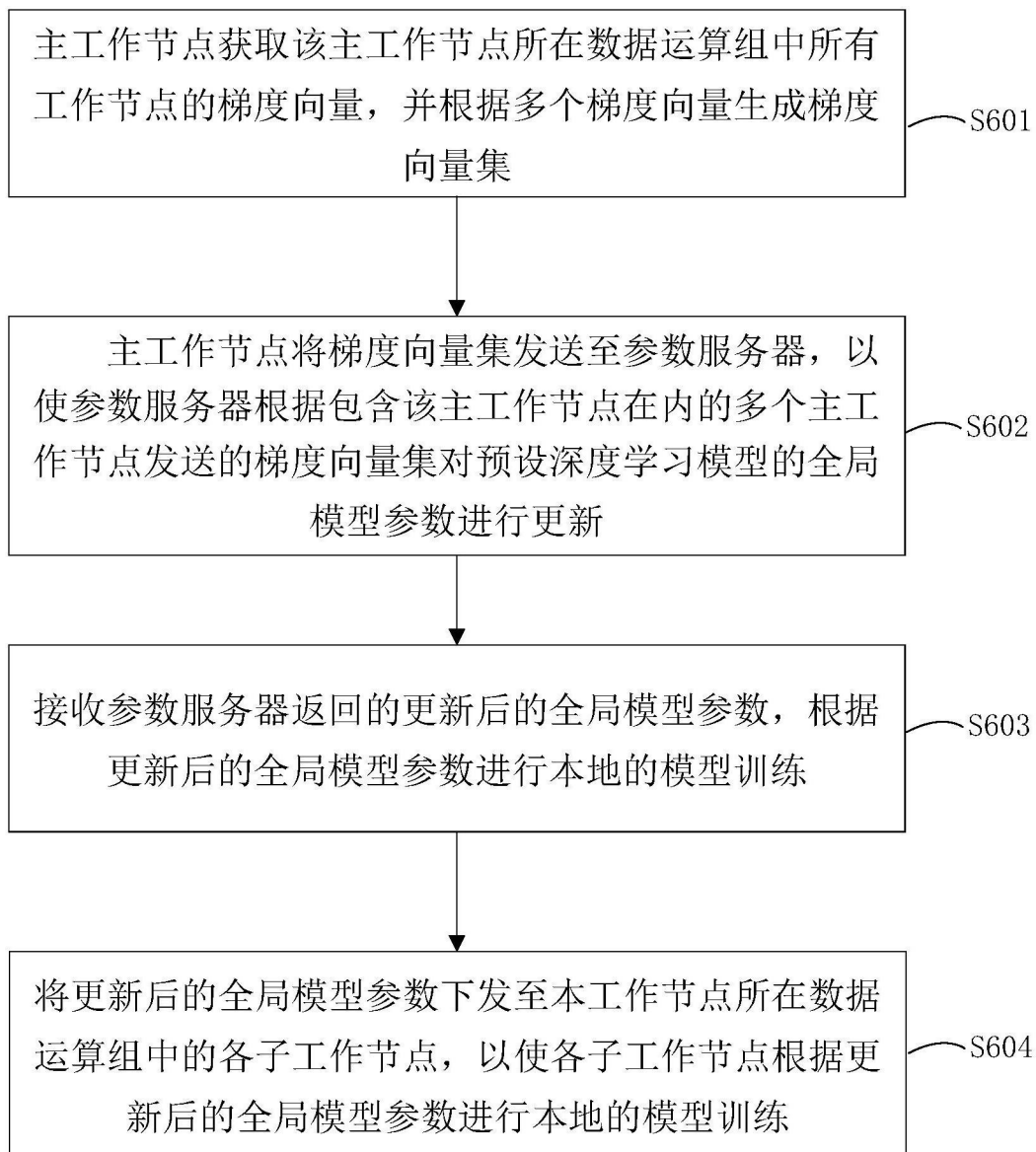


图6

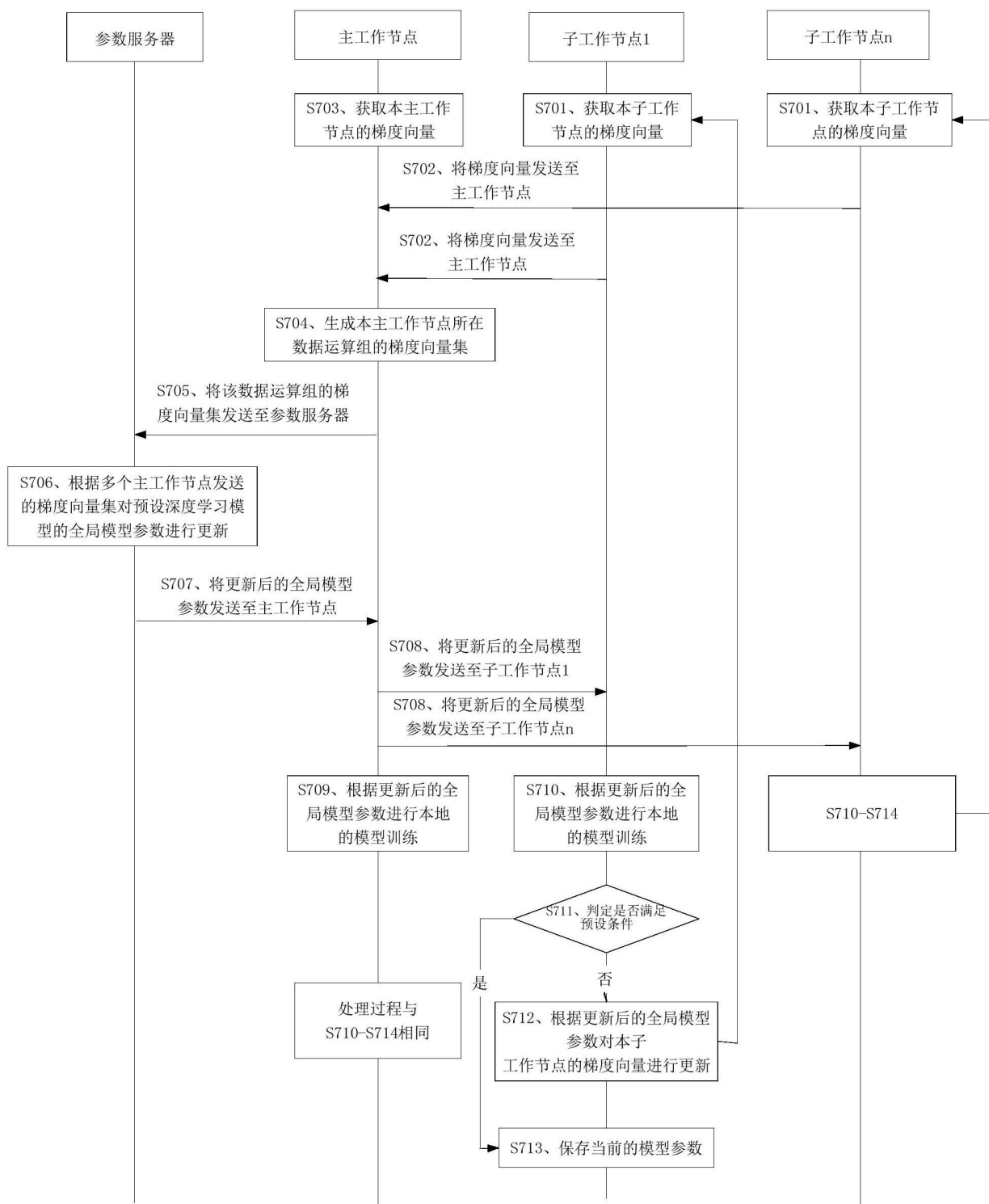


图7

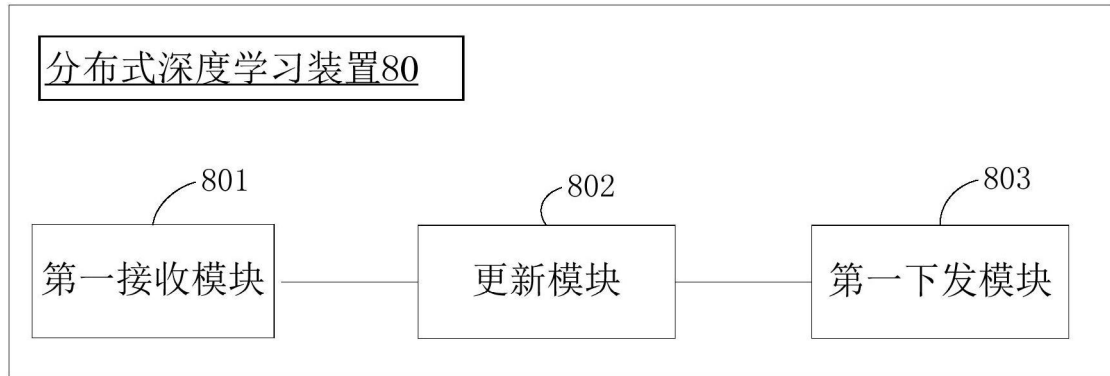


图8

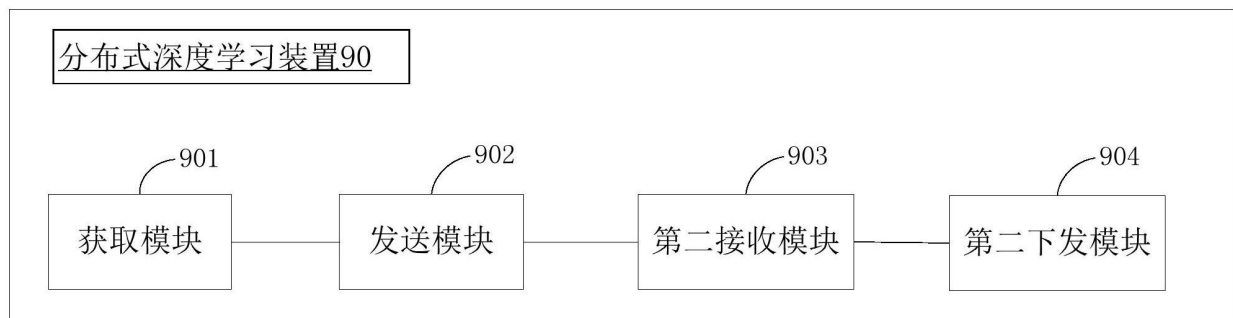


图9

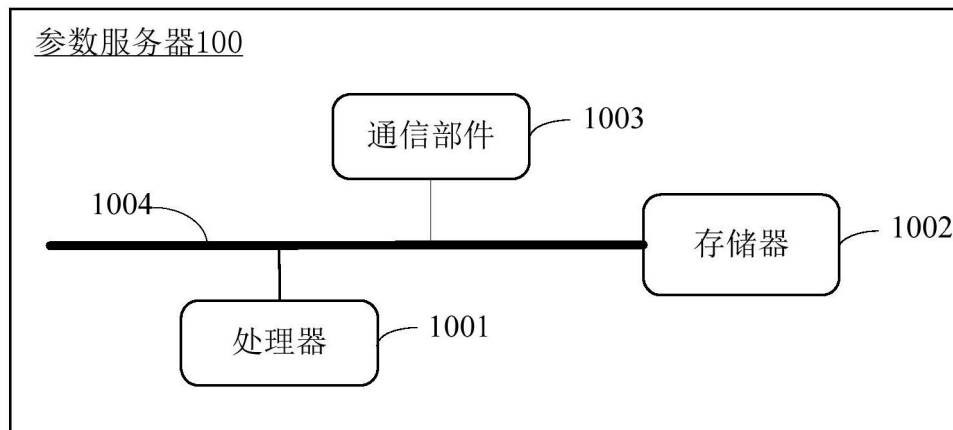


图10

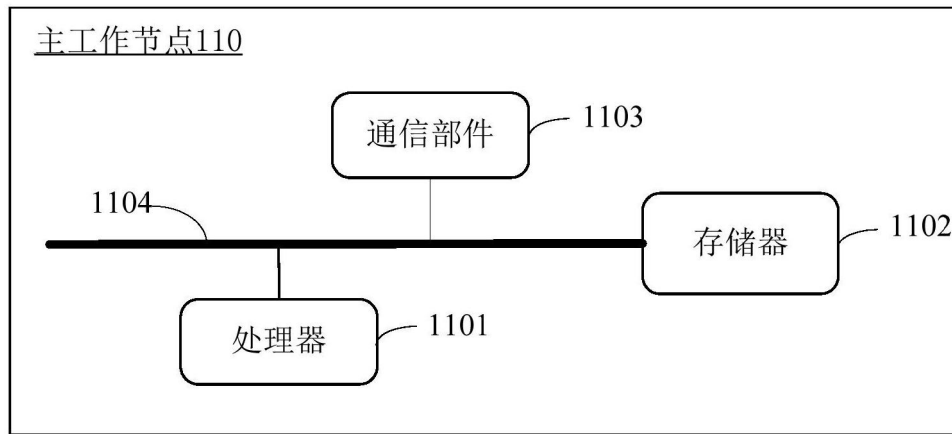


图11