



# (12) 发明专利

(10) 授权公告号 CN 110276074 B

(45) 授权公告日 2023.04.18

(21) 申请号 201910539080.0

G06F 16/33 (2019.01)

(22) 申请日 2019.06.20

G06F 18/214 (2023.01)

(65) 同一申请的已公布的文献号

G06F 18/241 (2023.01)

申请公布号 CN 110276074 A

G06N 20/00 (2019.01)

(43) 申请公布日 2019.09.24

### (56) 对比文件

(73) 专利权人 出门问问信息科技有限公司

CN 107924393 A, 2018.04.17

地址 100094 北京市海淀区中关村大街19

CN 104615425 A, 2015.05.13

号办公A楼10层1001

US 2006149553 A1, 2006.07.06

CN 104360994 A, 2015.02.18

(72) 发明人 王倪东 陈晓宇 林凤绿 李倩

审查员 胡平

(74) 专利代理机构 北京鼎承知识产权代理有限公司

11551

专利代理师 李伟波 韩德凯

(51) Int. Cl.

G06F 40/30 (2020.01)

G06F 16/35 (2019.01)

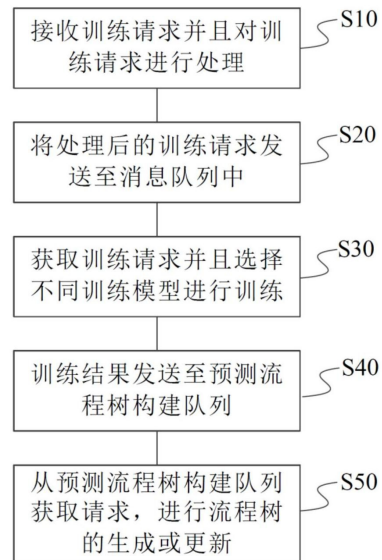
权利要求书2页 说明书10页 附图10页

### (54) 发明名称

自然语言处理的分布式训练方法、装置、设备及存储介质

### (57) 摘要

本公开提供了一种自然语言处理的分布式训练方法,包括:接收训练请求并且对训练请求进行处理;将包括训练数据及相应参数的处理后训练请求发送至消息队列中;从消息队列中获取训练请求,并且根据训练请求来选择不同的训练模型进行训练;将训练完成后的训练结果发送至预测流程树构建队列;以及从预测流程树构建队列中获取请求,进行流程树的生成或更新,并且上传生成的流程树或更新的流程树。本公开还提供了一种自然语言处理的分布式训练装置、电子设备及可读存储介质。



1. 一种自然语言处理的分布式训练方法,其特征在于,包括:
  - 接收训练请求并且对所述训练请求进行处理,所述训练请求中包括训练数据、词典数据和/或正则表达式数据;
  - 将包括所述训练数据及相应参数的处理后训练请求发送至消息队列中;
  - 从所述消息队列中获取所述训练请求,并且根据所述训练请求来选择不同的训练模型进行训练;
  - 将训练完成后的训练结果发送至预测流程树构建队列;以及
  - 从所述预测流程树构建队列中获取请求,进行流程树的生成或更新,并且上传生成的流程树或更新的流程树,所述流程树的更新包括修改所述流程树以及添加意图;
  - 在所述预测流程树生成或更新之后,进行配置上传处理;所述配置包括生成或更新之后的所述流程树的内部配置参数,基于所述内部配置参数确定训练类型。
2. 如权利要求1所述的方法,其特征在于,在训练服务器接收到所述训练请求后,根据所述训练请求的参数来构建训练参数,并且对所述训练参数进行解析以根据解析后的所述训练参数来选择相应的训练模型。
3. 如权利要求2中所述的方法,其特征在于,训练完成后,还包括:将训练后的训练模型及相应资源文件保存至对象存储中。
4. 如权利要求3所述的方法,其特征在于,还包括:更新训练后的训练模型的版本号,以及保存与所述训练相关的数据以便进行训练模型版本控制。
5. 如权利要求3或4所述的方法,其特征在于,还包括:将模型训练结果上传至配置中心,所述模型训练结果至少还包括训练后的所述训练模型的映射路径。
6. 如权利要求5所述的方法,其特征在于,还包括:当所述配置中心进行更新后,至少根据所述映射路径使得预测服务器从所述对象存储获取训练后的所述训练模型,以便更新预测服务器中的所述训练模型。
7. 一种自然语言处理的分布式训练装置,其特征在于,包括:
  - 训练请求接收模块,接收训练请求并且对所述训练请求进行处理,所述训练请求中包括训练数据、词典数据和/或正则表达式数据,将包括所述训练数据及相应参数的处理后训练请求发送至消息队列中;
  - 训练服务器模块,从所述消息队列中获取所述训练请求,并且根据所述训练请求来选择不同的训练模型进行训练,训练完成后将训练结果发送至预测流程树构建队列;以及
  - 配置包装模块,从所述预测流程树构建队列中获取请求,进行流程树的生成或更新,并且上传生成的流程树或更新的流程树,所述流程树的更新包括修改所述流程树以及添加意图;
  - 在所述预测流程树生成或更新之后,进行配置上传处理;所述配置包括生成或更新之后的所述流程树的内部配置参数,基于所述内部配置参数确定训练类型。
8. 一种电子设备,其特征在于,包括:
  - 存储器,所述存储器存储执行指令;以及
  - 处理器,所述处理器执行所述存储器存储的执行指令,使得所述处理器执行如权利要求1至6中任一项所述的方法。
9. 一种计算机可读存储介质,其特征在于,所述可读存储介质中存储有执行指令,所述

执行指令被处理器执行时用于实现如权利要求1至6中任一项所述的方法。

## 自然语言处理的分布式训练方法、装置、设备及存储介质

### 技术领域

[0001] 本公开涉及一种自然语言处理的分布式训练方法、自然语言处理的分布式训练装置、电子设备及计算机可读存储介质。

### 背景技术

[0002] 目前,自然语言处理工作都是由自然语言处理工程师来从项目经理手上获取数据并进行预处理训练出模型并完成部分的代码再交付给项目经理进行后续工程步骤,但是这个阶段项目经理对模型调整以及训练过程都是未知,会造成自然语言处理工程师在评估测试时效果很好,但是训练自然语言处理模型的实际效果却很差。

[0003] 因此,如何让非专业人士通过对自身数据理解就能完成自然语言处理工程师常用训练以及调优工作,是需要解决的一个问题。

### 发明内容

[0004] 为了解决上述技术问题中的至少一个,本公开提供了一种自然语言处理的分布式训练方法、自然语言处理的分布式训练装置、电子设备及计算机可读存储介质。

[0005] 根据本公开的一个方面,一种自然语言处理的分布式训练方法,包括:接收训练请求并且对所述训练请求进行处理;将包括训练数据及相应参数的处理后训练请求发送至消息队列中;从所述消息队列中获取所述训练请求,并且根据所述训练请求来选择不同的训练模型进行训练;将训练完成后的训练结果发送至预测流程树构建队列;以及从所述预测流程树构建队列中获取请求,进行流程树的生成或更新,并且上传生成的流程树或更新的流程树。

[0006] 根据本公开的至少一个实施方式,在训练服务器接收到所述训练请求后,根据所述训练请求的参数来构建训练参数,并且对所述训练参数进行解析以根据解析后的所述训练参数来选择相应的训练模型。

[0007] 根据本公开的至少一个实施方式,所述训练请求中包括训练数据、词典数据和/或正则表达式数据。

[0008] 根据本公开的至少一个实施方式,训练完成后,还包括:将训练后的训练模型及相应资源文件保存至对象存储中。

[0009] 根据本公开的至少一个实施方式,还包括:更新训练后的训练模型的版本号,以及保存与所述训练相关的数据以便进行训练模型版本控制。

[0010] 根据本公开的至少一个实施方式,还包括:将模型训练结果上传至配置中心,所述模型训练结果至少还包括训练后的所述训练模型的映射路径。

[0011] 根据本公开的至少一个实施方式,还包括:当所述配置中心进行更新后,至少根据所述映射路径使得预测服务器从所述对象存储获取训练后的所述训练模型,以便更新预测服务器中的所述训练模型。

[0012] 根据本公开的另一方面,一种自然语言处理的分布式训练装置,包括:训练请求接

收模块,接收训练请求并且对所述训练请求进行处理,将包括训练数据及相应参数的处理后训练请求发送至消息队列中;训练服务器模块,从所述消息队列中获取所述训练请求,并且根据所述训练请求来选择不同的训练模型进行训练,训练完成后将训练结果发送至预测流程树构建队列;以及配置包装模块,从所述预测流程树构建队列中获取请求,进行流程树的生成或更新,并且上传生成的流程树或更新的流程树。

[0013] 根据本公开的再一方面,一种电子设备,包括:存储器,所述存储器存储执行指令;以及处理器,所述处理器执行所述存储器存储的执行指令,使得所述处理器执行如上所述的方法。

[0014] 根据本公开的又一方面,一种计算机可读存储介质,所述可读存储介质中存储有执行指令,所述执行指令被处理器执行时用于实现如上所述的方法。

### 附图说明

[0015] 附图示出了本公开的示例性实施方式,并与其说明一起用于解释本公开的原理,其中包括了这些附图以提供对本公开的进一步理解,并且附图包括在本说明书中并构成本说明书的一部分。

[0016] 图1是根据本公开一个实施方式的自然语言处理的分布式训练方法的流程图。

[0017] 图2是根据本公开一个实施方式的自然语言处理的分布式训练处理的时序图。

[0018] 图3是根据本公开一个实施方式的自然语言处理的分布式训练系统的框架图。

[0019] 图4是根据本公开一个实施方式的训练方法的流程图。

[0020] 图5是根据本公开一个实施方式的流程树的生成或更新的方法的流程图。

[0021] 图6是根据本公开一个实施方式的配置上传方法的流程图。

[0022] 图7是根据本公开一个实施方式的自然语言处理的分布式训练装置的示意性视图。

[0023] 图8是根据本公开一个实施方式的训练装置的示例性框图。

[0024] 图9是根据本公开一个实施方式的流程树的生成或更新装置的示意性框图。

[0025] 图10是根据本公开一个实施方式的配置上传装置的示意性框图。

[0026] 图11是根据本公开的一个实施方式的电子设备的示意性视图。

### 具体实施方式

[0027] 下面结合附图和实施方式对本公开作进一步的详细说明。可以理解的是,此处所描述的具体实施方式仅用于解释相关内容,而非对本公开的限定。另外还需要说明的是,为了便于描述,附图中仅示出了与本公开相关的部分。

[0028] 需要说明的是,在不冲突的情况下,本公开中的实施方式及实施方式中的特征可以相互组合。下面将参考附图并结合实施方式来详细说明本公开。

[0029] 本公开的技术方案可以针对语义理解中的意图分类、或语义槽填充等问题,可以实现由用户自定义训练语料和词典等资源数据,系统根据这些数据进行一键训练、一键发布或一键回滚等功能。例如,意图分类任务通常是一个多分类问题,在本公开中可以将多分类问题转化为多个二分类问题,每一个意图都是一个二分类任务,每个二分类任务单独训练一个模型,最终的多分类结果是概率值最高的意图二分类结果;而对于语义槽填充而言,

其本身是一个序列标注的问题,对于每一个意图设计内的一批语义槽标记则训练一个序列标注模型。此外,虽然本公开中可能以意图分类或语义槽填充为例进行说明,但是需要注意的是,本公开的技术方案并不限于此,其可以扩展到通用机器学习平台。

[0030] 根据本公开的一个实施方式,提供了一种自然语言处理的分布式训练方法,包括:接收训练请求并且对训练请求进行处理;将包括训练数据及相应参数的处理后训练请求发送至消息队列中;从消息队列中获取训练请求,并且根据训练请求来选择不同的训练模型进行训练;训练完成后将训练结果发送至预测流程树构建队列;以及从预测流程树构建队列中获取请求,进行流程树的生成或更新,并且上传生成的流程树或更新的流程树。

[0031] 下面,将参照图1来描述该训练方法。如图1所示,该方法可以包括步骤S10:接收训练请求并且对训练请求进行处理;步骤S20:将处理后的训练请求发送至消息队列中;步骤S30:获取处理后的训练请求并且选择不同训练模型进行训练;步骤S40:将训练结果发送至预测流程树构建队列;以及步骤S50:从预测流程树构建队列中获取请求,进行流程树的生成与更新。

[0032] 在步骤S10中,接收训练请求并且对训练请求进行处理。在该步骤中,用户可以提交训练请求,然后训练请求可以被进行处理,例如,可以包括消息处理层等,通过该消息处理层来对训练请求进行处理。在该步骤中,可以在接收训练请求后,可以根据训练请求参数来构建训练参数,并且对训练数据的字段来进行解析,其中可以包括对通过训练请求直接传入的JSON数据或者数据源进行结构化处理等。

[0033] 作为一个示例,图2示出了训练处理的时序图。在图2中,通过POST请求将训练请求传递给训练控制器,训练控制器用于训练请求的接收,其中该POST请求中带有数据信息,该数据信息可以包括训练数据、词典数据和/或正则表达式数据等。

[0034] 然后验证输入配置及数据,该输入配置可以是参数对应的训练的配置等,该数据可以是训练数据、词典数据和/或正则表达式数据、或者它们的URL等。

[0035] 然后过滤器可以对请求进行过滤,之后解析配置生成对应的训练信息,例如可以根据URL将对应的数据下载并且存储到对应位置,且进行数据组装。然后训练预服务可以连接数据库将数据等发送给消息队列。

[0036] 此外,训练预服务、过滤器、训练控制器可以返回响应信息,这些响应信息例如可以包括能够查询到的训练标识、成功失败标识等,其可以为查询ID。

[0037] 在步骤S20中,将包括训练数据及相应参数的处理后训练请求发送至消息队列中。例如在图2中,在训练预服务完成后将训练请求发送至消息队列中。

[0038] 在步骤S30中从消息队列中获取训练请求,并且根据训练请求来选择不同的训练模型进行训练。例如可以通过后台线程从请求队列中获取训练请求。例如在图2中,训练消费者(训练服务器)从消息队列中取出训练请求并且反序列化,例如存入消息队列中的格式需要转成程序内的类加载格式等。

[0039] 在根据训练请求选取不同的训练模型进行训练时,可以根据训练参数来选择合适的模型,调用具体的训练流程,然后进行训练脚本的调用,其中每一步脚本执行异常处理,整个流程可以为一个事务。

[0040] 在步骤S40中,训练完成后将训练结果发送至预测流程树构建队列。其中可以在训练服务器中组织预测配置生成参数,并且将预测配置参数发送至消息队列(预测流程树构

建队列)中进行存储。

[0041] 在步骤S50中,从预测流程树构建队列中获取请求,进行流程树的生成或更新,并且上传生成的流程树或更新的流程树。可选地,在从预测流程树构建队列中获取请求,然后解析参数进行参数判断,并在处理相同流程树利用公共缓存做加锁处理,最后进行流程树的生成或更新。在流程树的生成或更新完成后,可以上传生成的流程树或更新的流程树,例如上传至配置中心等。

[0042] 根据本公开的一个可选实施例,可以在步骤S10:接收训练请求并且对训练请求进行处理之后,且在步骤S20:将处理后的训练请求发送至消息队列中之前,可以进行状态存储,例如可以为训练请求记录表的形式等。并且在步骤S20将处理后的训练请求发送至消息队列中之后,可以返回请求。

[0043] 下面将参照图3来描述根据本公开的整体训练方式。图3是根据本公开的一个实施方式的训练系统的框架图。

[0044] 用户100可以发送训练请求,例如通过图2所示的POST请求(带有数据信息,该数据信息可以包括训练数据、词典数据和/或正则表达式数据等),用户100的训练请求可以进入消息队列200(例如用于异步地进行消息处理的存放以便提供给训练服务器300)中。

[0045] 根据本公开的方法,在用户100与消息队列200之间可以包括消息处理层,该消息处理层可以用于进行如图2所示的下述处理:例如,验证输入配置及数据,通过过滤器对训练请求进行过滤,解析配置生成对应的训练信息等。

[0046] 训练请求经过消息队列200后可以进入训练服务器300中,训练服务器300可以对训练请求进行解析,并且根据参数(例如预定义的JSON形式参数等)来在训练服务器300中选择不同的训练模型进行训练。

[0047] 训练完成后,模型结果文件(例如模型的实体文件)及对应的资源文件(例如正则表达式、词典、语义内容、原特征提取的中间处理结果等等)可以被打包上传,保存至对象存储400中。

[0048] 同时,训练完成后,更新本次模型的版本号,将本次训练相关的数据(例如对应的版本号等版本识别的信息、模型对应的映射路径(指向对象存储400中存储的模型实体)等)可以保存至数据库600中以便进行模型版本控制,例如为了回滚、模型的复用加载等。

[0049] 消息队列500用于进行版本的统一管理,例如在进行高并发时,通过消息队列500可以避免读写的脏读或幻读等情况的发生,从而避免版本管理混乱。

[0050] 同时,训练完成后,模型训练结果文件(例如,模型对应的映射路径、加载所需要的信息等)上传至配置中心700。

[0051] 预测服务器800监听配置中心700的配置是否被改变,如果被改变,预测服务器800则收到相应的被改变的请求,如果配置有更新时,则自动拉取对应的模型并实时进行线上的模型更新。

[0052] 这样,预测用户900可以根据预测服务器800中的模型来进行预测等。

[0053] 根据本公开的一个具体实施例,提供了一种训练方法。如图4所示,该方法可以包括步骤S41:根据训练请求参数来构建训练参数;步骤S42:对训练数据字段进行解析;步骤S43:对训练数据进行特征提取;步骤S44:根据选择模型来对Embedding/格式进行转换;步骤S45:将训练数据分为训练集及验证集;步骤S46:调用底层算法进行训练;步骤S47:对训

练结果进行验证。

[0054] 在步骤S41中,根据训练请求参数来构建训练参数。

[0055] 在步骤S42中,对训练数据字段进行解析,该解析可以包括请求直接传入的JSON数据或者数据源进行结构化处理,例如该结构化处理可以根据预定义形式。

[0056] 在步骤S43中,对训练数据进行特征提取。对于提取特征的选择,可以根据实际需要来选择所需特征。

[0057] 在步骤S44中,根据选择模型来对Embedding/格式进行转换,以形成模型所需的形式。

[0058] 在步骤S45中,将训练数据分为训练集及验证集,其中该训练集数据用于对所选训练模型进行训练,而验证集数据则用于对训练后的模型进行验证。

[0059] 在步骤S46中,调用底层算法使用训练集数据进行训练。该底层算法可以是已有的机器学习算法等。

[0060] 在步骤S47中,在训练完成后,可以使用验证集数据来对训练模型的训练结果进行验证。

[0061] 根据本公开的一个具体实施例,将具体描述在上述步骤S50中进行流程树的生成或更新的具体示例。其中,该示例以意图识别为例进行说明,但是本领域的技术人员应当理解,该示例也可以用于其它问题。

[0062] 下面将参照图5来详细的描述。流程树的生成或更新的方法包括以下步骤。

[0063] 步骤S51:根据相应的选择,传入指定的流程树,并且查询是否存在该流程树。在不存在的的情况下,则进行步骤S58(流程树新增操作),如果在存在的情况下,则进行步骤S52(流程树修改操作)。

[0064] 首先,针对流程树修改操作来进行描述。当在步骤S51中,判断不存在相应的流程树的情况下,在步骤S52中,判断该流程树是否为在原流程树上的修改。该修改分为两种情况:更新及新增。

[0065] 在步骤S54中,示出了新增的修改方式,也就是说在流程树上增加对应的数据转换模块及触发器,并且在对应汇总的数据转换模块中进行相应意图的添加。

[0066] 在步骤S53中,示出了更新的修改方式,在该方式中,根据传入的算法类型及解析操作的意图,对流程树中的算法类型做比对校验。

[0067] 在步骤S53与S54之后,在步骤S55中,在指定的数据转换模块中获取匹配是否有对应意图的判断。如果有,则进行相同算法不同版本更新,如果无,则进行算法更新。

[0068] 步骤S56中,示出了进行相同算法不同版本更新的方式,通过操作找到对应的意图的数据转换模块,修改相应模型及NLP(自然语言处理)资源路径。

[0069] 在步骤S57中,则示出了算法更新的具体方式,在该方式中,删除原有意图的数据转换模块及触发器,完成新增意图的数据转换模块及触发器的添加操作。

[0070] 而对于步骤S58的流程树新增操作而言,在步骤S58中,进行新增操作,完成构建新的预测流程树的过程,其中,新增对应的数据转换模块及触发器等。

[0071] 此外,根据本公开的技术方案,在预测流程树生成或更新之后,则进行配置上传的处理。

[0072] 下面参照图6来描述该配置上传的方法。



[0073] 在步骤S61中,首先获取生成或更新的流程树的内部配置参数。

[0074] 在步骤S62中,判断训练类型,其中,根据获取的流程树的内部配置参数来判断该训练类型。

[0075] 在步骤S63中,判断配置中心的配置空间(name space)是否存在。其中不同任务种类会有不同的中控控制,配置空间为作用于任务上的一个配置块,不同任务种类具有不同的配置块。如果没有配置空间,则新增。

[0076] 在步骤S64中,修改和/或添加配置空间字段。

[0077] 在步骤S65中,判断传入任务的配置空间是否存在。如果不存在,则新增。

[0078] 在步骤S66中,修改和插入任务配置空间。

[0079] 在步骤S67中,进行发布。在上面的操作完成,进行发布生效,以将所有的配置均同步等。

[0080] 根据本公开的上述技术方案,可以实现每个任务的不同模型版本均是可控的,并且对于分布式训练请求及后续处理,可以同时进行多个任务的训练,并且在训练出错时也不会影响其它正常的训练,不会影响到线上服务。此外,在训练完成后不需要重新写一遍推断流程,可以实时生效到预测服务器上。

[0081] 在本公开中,可以支持不同协议的对象存储,包括私有/公有分布式对象存储、网络文件本地及本地文件系统等。并且本公开的技术方案可以扩展到通用机器学习平台功能。

[0082] 根据本公开的另一实施方式,提供了一种自然语言处理的分布式训练装置。

[0083] 如图7所示,该分布式训练装置700可以包括训练请求接收模块701、训练服务器模块702、及配置包装模块703。

[0084] 训练请求接收模块701接收训练请求并且对所述训练请求进行处理,将包括训练数据及相应参数的处理后训练请求发送至消息队列中;训练服务器模块702从所述消息队列中获取所述训练请求,并且根据所述训练请求来选择不同的训练模型进行训练,训练完成后将训练结果发送至预测流程树构建队列;以及配置包装模块703从所述预测流程树构建队列中获取请求,进行流程树的生成或更新,并且上传生成的流程树或更新的流程树

[0085] 作为具体示例,训练请求接收模块701接收训练请求并且对训练请求进行处理。在该模块中,用户可以提交训练请求,然后训练请求可以被进行处理,例如,可以包括消息处理层等,通过该消息处理层来对训练请求进行处理。在该模块中,可以在接收训练请求后,可以根据训练请求参数来构建训练参数,并且对训练数据的字段来进行解析,其中可以包括对通过训练请求直接传入的JSON数据或者数据源进行结构化处理等。其中,可以如图2所示,通过POST请求将训练请求传递给训练控制器,训练控制器用于训练请求的接收,其中该POST请求中带有数据信息,该数据信息可以包括训练数据、词典数据和/或正则表达式数据等。然后验证输入配置及数据,该输入配置可以是参数对应的训练的配置等,该数据可以是训练数据、词典数据和/或正则表达式数据、或者它们的URL等。然后过滤器可以对请求进行过滤,之后解析配置生成对应的训练信息,例如可以根据URL将对应的数据下载并且存储到对应位置,且进行数据组装。然后训练预服务可以连接数据库将数据等发送给消息队列。此外,训练预服务、过滤器、训练控制器可以返回响应信息,这些响应信息例如可以包括能够查询到的训练标识、成功失败标识等,其可以为查询ID。将包括训练数据及相应参数的处理

后训练请求发送至消息队列中。例如在图2中,在训练预服务完成后将训练请求发送至消息队列中。

[0086] 训练服务器模块702从消息队列中获取训练请求,并且根据训练请求来选择不同的训练模型进行训练。例如可以通过后台线程从请求队列中获取训练请求。例如在图2中,训练消费者(训练服务器)从消息队列中取出训练请求并且反序列化,例如存入消息队列中的格式需要转成程序内的类加载格式等。

[0087] 在根据训练请求选取不同的训练模型进行训练时,可以根据训练参数来选择合适的模型,调用具体的训练流程,然后进行训练脚本的调用,其中每一步脚本执行异常处理,整个流程可以为一个事务。训练完成后将训练结果发送至预测流程树构建队列。其中可以在训练服务器中组织预测配置生成参数,并且将预测配置参数发送至消息队列(预测流程树构建队列)中进行存储。

[0088] 配置包装模块703从预测流程树构建队列中获取请求,进行流程树的生成或更新,并且上传生成的流程树或更新的流程树。可选地,在从预测流程树构建队列中获取请求,然后解析参数进行参数判断,并在处理相同流程树利用公共缓存做加锁处理,最后进行流程树的生成或更新

[0089] 根据本公开的一个可选实施例,可以在训练请求接收模块701中接收训练请求并且对训练请求进行处理之后且在将处理后的训练请求发送至消息队列中之前,可以进行状态存储,例如可以为训练请求记录表的形式等。

[0090] 根据本公开的进一步实施方式,分布式训练装置还可以包括对象存储模块,训练完成后,训练后的训练模型及相应资源文件被保存至对象存储中。

[0091] 根据本公开的进一步实施方式,分布式训练装置还可以包括版本控制模块,其中,更新训练后的训练模型的版本号,以及保存与所述训练相关的数据以便进行训练模型版本控制。

[0092] 根据本公开的进一步实施方式,分布式训练装置还可以包括配置中心模块,该模块接收模型训练结果并保存,该模型训练结果至少还包括训练后的训练模型的映射路径。

[0093] 根据本公开的一个具体实施例,提供了一种训练装置。如图8所示,该装置800可以包括以下模块:

[0094] 构建模块801:根据训练请求参数来构建训练参数。

[0095] 解析模块802:对训练数据字段进行解析,该解析可以包括请求直接传入的JSON数据或者数据源进行结构化处理,例如该结构化处理可以根据预定义形式。

[0096] 提取模块803:对训练数据进行特征提取,对于提取特征的选择,可以根据实际需要来选择所需特征。

[0097] 转换模块804:根据选择模型来对Embedding/格式进行转换,以形成模型所需的形式。

[0098] 划分模块805:将训练数据分为训练集及验证集,其中该训练集数据用于对所选训练模型进行训练,而验证集数据则用于对训练后的模型进行验证。

[0099] 训练模块806:调用底层算法进行训练,该底层算法可以是已有的机器学习算法等。

[0100] 验证模块807:对训练结果进行验证,在训练完成后,可以使用验证集数据来对训

练模型的训练结果进行验证。

[0101] 此外,在对训练结果进行验证完成后,可以将结果打包上传至对象存储模块中。

[0102] 根据本公开的再一实施方式,提供了一种流程树的生成或更新的装置。

[0103] 下面将参照图9来详细的描述。流程树的生成或更新的装置900可以包括以下模块:

[0104] 查询模块901:根据相应的选择,传入指定的流程树,并且查询是否存在该流程树。在不存在的情况下,则进入新增模块903(流程树新增操作),如果在存在的情况下,则进行修改模块902(流程树修改操作)。

[0105] 当在查询模块901中,判断不存在相应的流程树的情况下,则修改模块902判断该流程树是否为在原流程树上的修改。该修改分为两种情况:更新及新增。对于新增的修改方式,也就是说在流程树上增加对应的数据转换模块及触发器,并且在对应汇总的数据转换模块中进行相应意图的添加。对于更新的修改方式,在该方式中,根据传入的算法类型及解析操作的意图,对流程树中的算法类型做比对校验。之后,在指定的数据转换模块中获取匹配是否有对应意图的判断。如果有,则进行相同算法不同版本更新,如果无,则进行算法更新。对于相同算法不同版本更新的方式,通过操作找到对应的意图的数据转换模块,修改相应模型及NLP(自然语言处理)资源路径。对于算法更新的具体方式,在该方式中,删除原有意图的数据转换模块及触发器,完成新增意图的数据转换模块及触发器的添加操作。

[0106] 在新增模块903中,进行新增操作,完成构建新的预测流程树的过程,其中,新增对应的数据转换模块及触发器等。

[0107] 此外,根据本公开的技术方案,在预测流程树生成或更新之后,则进行配置上传的处理,该配置上传处理可以通过配置上传装置来实现。

[0108] 下面参照图10来描述该配置上传装置,该配置上传装置可以包括:

[0109] 参数获取模块1001,首先获取生成或更新的流程树的内部配置参数。

[0110] 类型判断模块1002,判断训练类型,其中,根据获取的流程树的内部配置参数来判断该训练类型。

[0111] 第一判断模块1003,判断配置中心的配置空间(name space)是否存在。其中不同任务种类会有不同的中控控制,配置空间为作用于任务上的一个配置块,不同任务种类具有不同的配置块。如果没有配置空间,则新增。

[0112] 字段处理模块1004,修改和/或添加配置空间字段。

[0113] 第二判断模块1005,判断传入任务的配置空间是否存在。如果不存在,则新增。

[0114] 空间处理模块1006,修改和插入任务配置空间。

[0115] 发布模块1007,进行发布。在上面模块的操作完成,进行发布生效,以将所有的配置均同步等。

[0116] 本公开还提供一种电子设备,如图11所示,该设备包括:通信接口1000、存储器2000和处理器3000。通信接口1000用于与外界设备进行通信,进行数据交互传输。存储器2000内存储有可在处理器3000上运行的计算机程序。处理器3000执行所述计算机程序时实现上述实施方式中方法。所述存储器2000和处理器3000的数量可以为一个或多个。

[0117] 存储器2000可以包括高速RAM存储器,也可以还包括非易失性存储器(non-volatile memory),例如至少一个磁盘存储器。

[0118] 如果通信接口1000、存储器2000及处理器3000独立实现,则通信接口1000、存储器2000及处理器3000可以通过总线相互连接并完成相互间的通信。所述总线可以是工业标准体系结构(ISA,Industry Standard Architecture)总线、外部设备互连(PCI,Peripheral Component)总线或扩展工业标准体系结构(EISA,Extended Industry Standard Component)总线等。所述总线可以分为地址总线、数据总线、控制总线等。为便于表示,该图中仅用一条粗线表示,但并不表示仅有一根总线或一种类型的总线。

[0119] 可选的,在具体实现上,如果通信接口1000、存储器2000、及处理器3000集成在一块芯片上,则通信接口1000、存储器2000、及处理器3000可以通过内部接口完成相互间的通信。

[0120] 流程图中或在此以其他方式描述的任何过程或方法描述可以被理解为,表示包括一个或更多个用于实现特定逻辑功能或过程的步骤的可执行指令的代码的模块、片段或部分,并且本公开的优选实施方式的范围包括另外的实现,其中可以不按所示出或讨论的顺序,包括根据所涉及的功能按基本同时的方式或按相反的顺序,来执行功能,这应被本公开的实施方式所属技术领域的技术人员所理解。处理器执行上文所描述的各个方法和处理。例如,本公开中的方法实施方式可以被实现为软件程序,其被有形地包含于机器可读介质,例如存储器。在一些实施方式中,软件程序的部分或者全部可以经由存储器和/或通信接口而被载入和/或安装。当软件程序加载到存储器并由处理器执行时,可以执行上文描述的方法中的一个或多个步骤。备选地,在其他实施方式中,处理器可以通过其他任何适当的方式(例如,借助于固件)而被配置为执行上述方法之一。

[0121] 在流程图中表示或在此以其他方式描述的逻辑和/或步骤,可以具体实现在任何可读存储介质中,以供指令执行系统、装置或设备(如基于计算机的系统、包括处理器的系统或其他可以从指令执行系统、装置或设备取指令并执行指令的系统)使用,或结合这些指令执行系统、装置或设备而使用。

[0122] 就本说明书而言,“可读存储介质”可以是任何可以包含、存储、通信、传播或传输程序以供指令执行系统、装置或设备或结合这些指令执行系统、装置或设备而使用的装置。可读存储介质的更具体的示例(非穷尽性列表)包括以下:具有一个或多个布线的电连接部(电子装置),便携式计算机盘盒(磁装置),随机存取存储器(RAM),只读存储器(ROM),可擦除可编程只读存储器(EPROM或闪速存储器),光纤装置,以及便携式只读存储器(CDROM)。另外,可读存储介质甚至可以是可在其上打印所述程序的纸或其他合适的介质,因为可以例如通过对纸或其他介质进行光学扫描,接着进行编辑、解译或必要时以其他合适方式进行处理来以电子方式获得所述程序,然后将其存储在存储器中。

[0123] 应当理解,本公开的各部分可以用硬件、软件或它们的组合来实现。在上述实施方式中,多个步骤或方法可以用存储在存储器中且由合适的指令执行系统执行的软件来实现。例如,如果用硬件来实现,和在另一实施方式中一样,可用本领域公知的下列技术中的任一项或他们的组合来实现:具有用于对数据信号实现逻辑功能的逻辑门电路的离散逻辑电路,具有合适的组合逻辑门电路的专用集成电路,可编程门阵列(PGA),现场可编程门阵列(FPGA)等。

[0124] 本技术领域的普通技术人员可以理解实现上述实施方式方法的全部或部分步骤是可以通程序来指令相关的硬件完成,所述的程序可以存储于一种可读存储介质中,该

程序在执行时,包括方法实施方式的步骤之一或其组合。

[0125] 此外,在本公开各个实施方式中的各功能单元可以集成在一个处理模块中,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模块中。上述集成的模块既可以采用硬件的形式实现,也可以采用软件功能模块的形式实现。所述集成的模块如果以软件功能模块的形式实现并作为独立的产品销售或使用,也可以存储在一个可读存储介质中。所述存储介质可以是只读存储器,磁盘或光盘等。

[0126] 在本说明书的描述中,参考术语“一个实施方式/方式”、“一些实施方式/方式”、“示例”、“具体示例”、或“一些示例”等的描述意指结合该实施方式/方式或示例描述的具体特征、结构、材料或者特点包含于本申请的至少一个实施方式/方式或示例中。在本说明书中,对上述术语的示意性表述不必须针对的是相同的实施方式/方式或示例。而且,描述的具体特征、结构、材料或者特点可以在任一个或多个实施方式/方式或示例中以合适的方式结合。此外,在不相互矛盾的情况下,本领域的技术人员可以将本说明书中描述的不同实施方式/方式或示例以及不同实施方式/方式或示例的特征进行结合和组合。

[0127] 此外,术语“第一”、“第二”仅用于描述目的,而不能理解为指示或暗示相对重要性或者隐含指明所指示的技术特征的数量。由此,限定有“第一”、“第二”的特征可以明示或者隐含地包括至少一个该特征。在本申请的描述中,“多个”的含义是至少两个,例如两个,三个等,除非另有明确具体的限定。

[0128] 本领域的技术人员应当理解,上述实施方式仅仅是为了清楚地说明本公开,而并非是对本公开的范围进行限定。对于所属领域的技术人员而言,在上述公开的基础上还可以做出其它变化或变型,并且这些变化或变型仍处于本公开的范围。

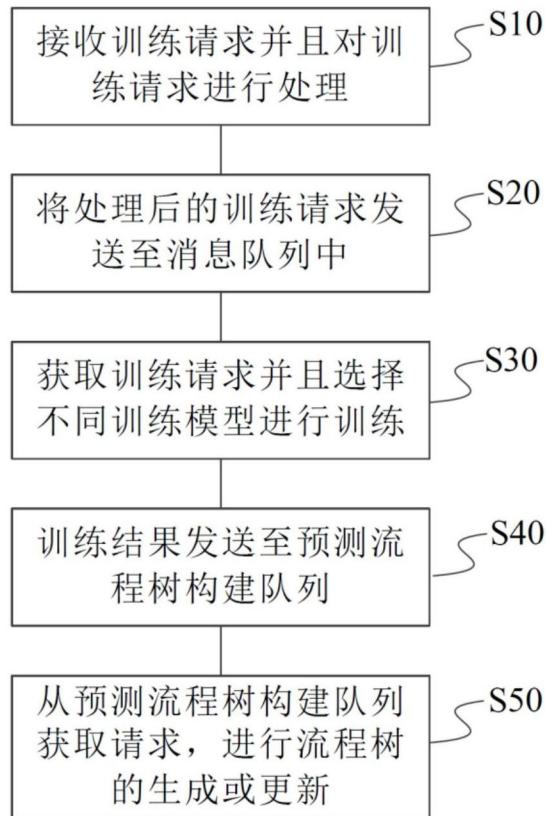


图1



图2

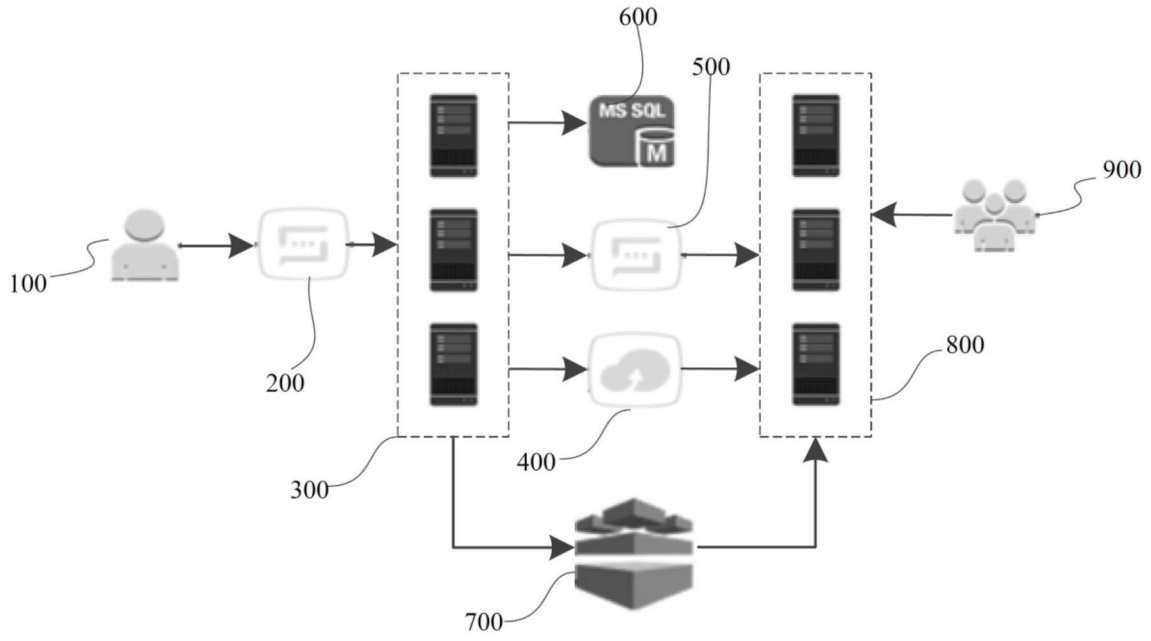


图3



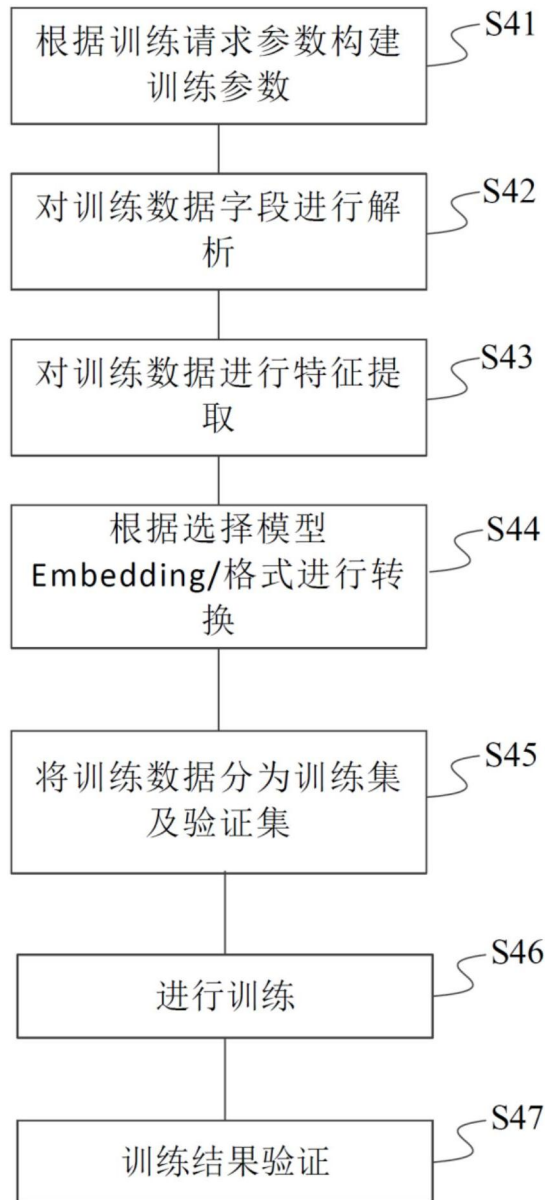


图4

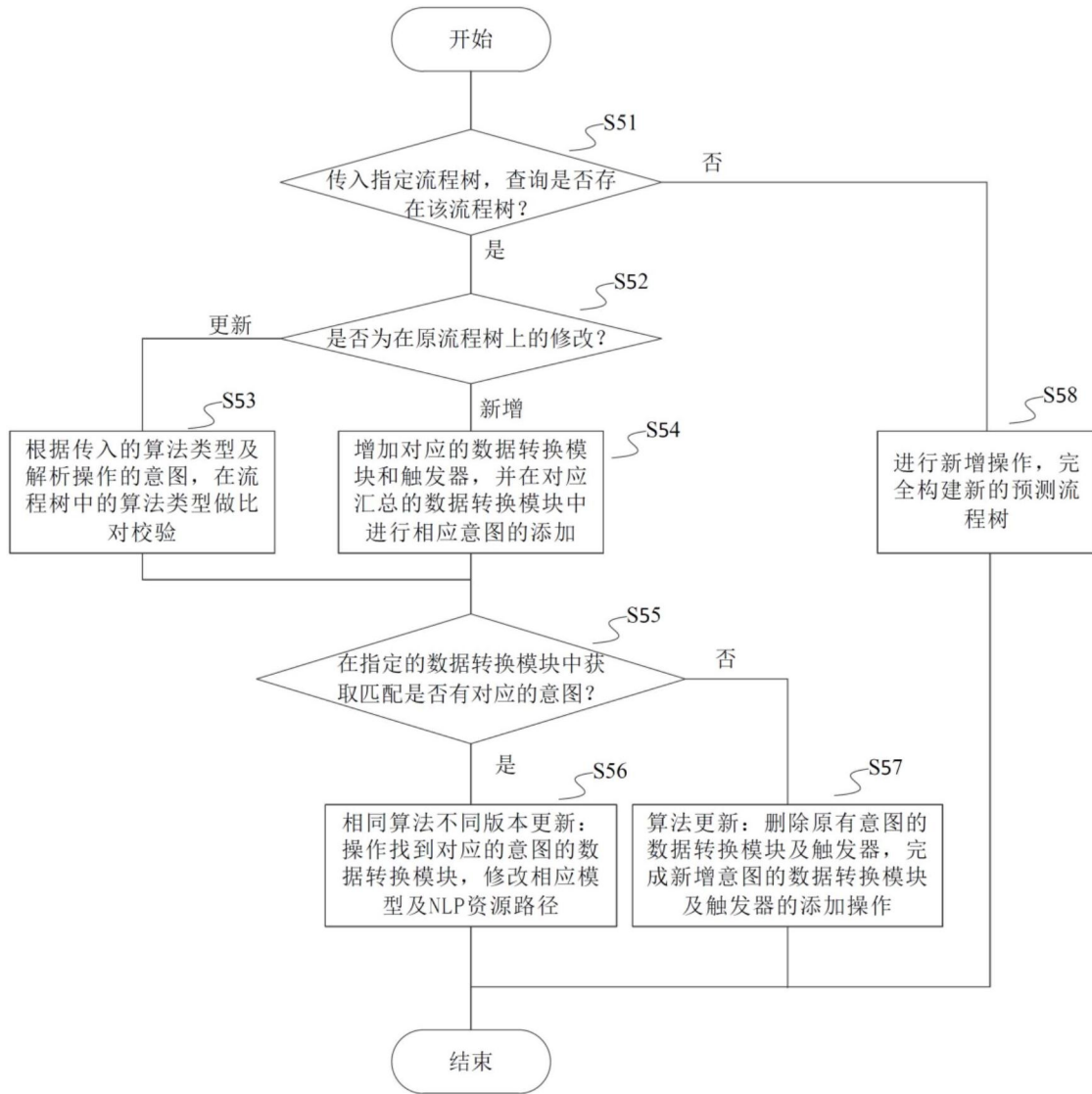


图5

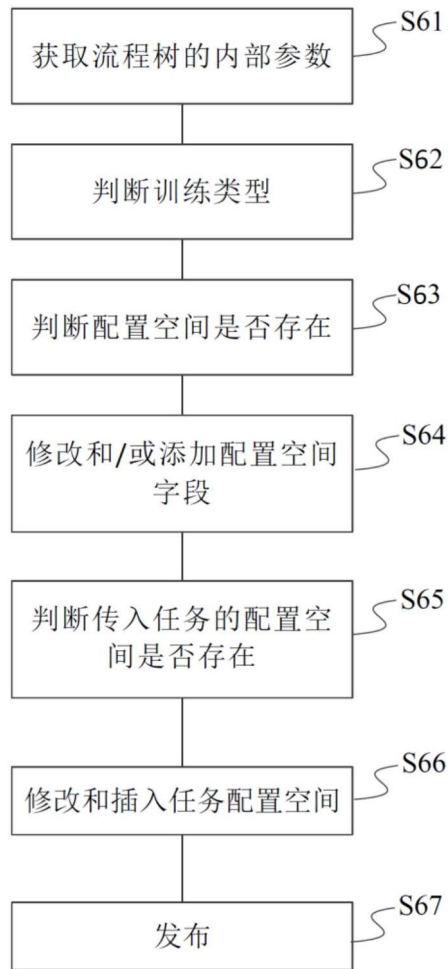


图6

**700**



图7

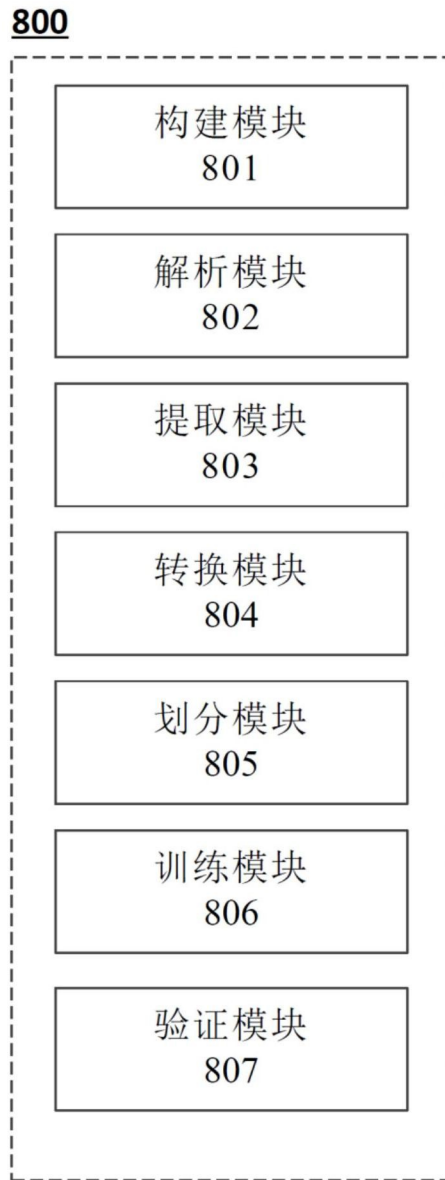


图8

**900**

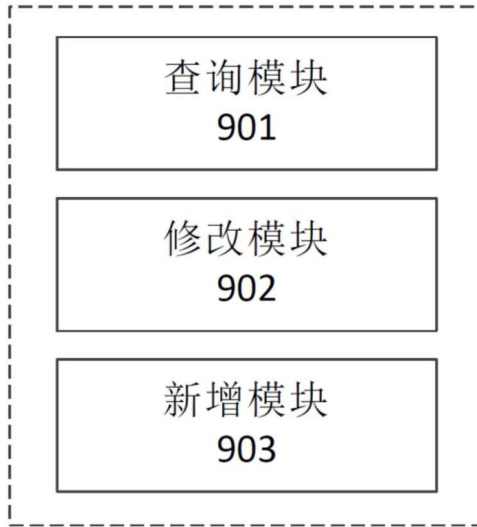


图9



图10

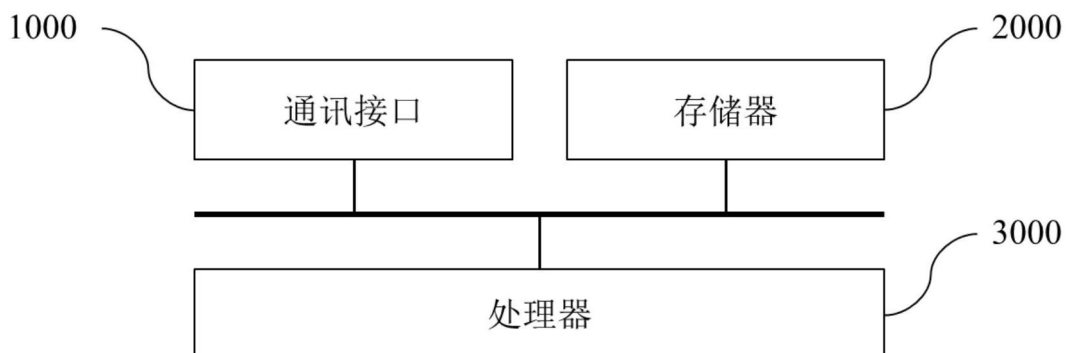


图11