(19)国家知识产权局

(12)发明专利申请

(10)申请公布号 CN 117332872 A
(43)申请公布日 2024.01.02

(21)申请号 202211097676.8

(22)申请日 2022.09.08

(71)申请人 北京富算科技有限公司
地址 100020 北京市朝阳区东三环中路9号19层2201

(72)发明人 尤志强 卞阳 赵东 朱崇炳 陈立峰

(74)专利代理机构 上海弼兴律师事务所 31283
专利代理师 罗朗 林嵩

(51)Int.Cl.
G06N 20/00(2019.01)

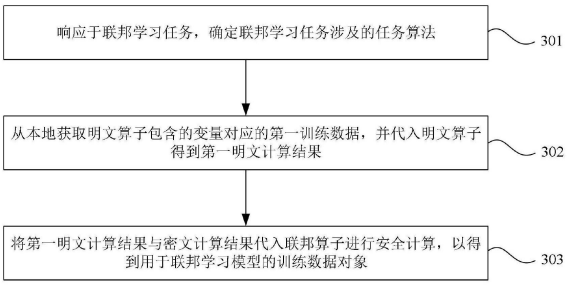
权利要求书3页 说明书12页 附图8页

(54)发明名称

纵向联邦学习模型的训练方法、装置、电子设备、介质

(57)摘要

本发明公开了纵向联邦学习模型的训练方法、装置、电子设备、介质。方法应用于多个参与方中的任一参与方，训练方法包括：响应于联邦学习任务，确定联邦学习任务涉及的任务算法，任务算法包括明文算子和联邦算子；从本地获取明文算子包含的变量对应的第一训练数据，并代入明文算子得到第一明文计算结果；将第一明文计算结果与密文计算结果代入联邦算子进行安全计算，以得到用于联邦学习模型的训练的数据对象；其中，密文计算结果为其他参与方提供的第二明文计算结果的密文形态，第二明文计算结果由其他参与方从其本地获取明文算子包含的变量对应的第二训练数据代入明文算子计算得到。本发明极大减少了通信次数和数量。



CN 117332872 A

1.一种纵向联邦学习模型的训练方法,其特征在于,应用于多个参与方中的任一参与方,所述训练方法包括:

响应于联邦学习任务,确定所述联邦学习任务涉及的任务算法,所述任务算法包括明文算子和联邦算子;

从本地获取所述明文算子包含的变量对应的第一训练数据,并代入所述明文算子得到第一明文计算结果;

将所述第一明文计算结果与密文计算结果代入所述联邦算子进行安全计算,以得到用于联邦学习模型的训练的数据对象;其中,所述密文计算结果为其他参与方提供的第二明文计算结果的密文形态,所述第二明文计算结果由所述其他参与方从其本地获取明文算子包含的变量对应的第二训练数据代入所述明文算子计算得到。

2.根据权利要求1所述的纵向联邦学习模型的训练方法,其特征在于,还包括:

响应于数据获取请求,从本地获取所述明文算子包含的变量对应的第三训练数据,并代入所述明文算子得到第三明文计算结果;

将所述第三明文计算结果的密文形态发送给所述其他参与方。

3.根据权利要求1所述的纵向联邦学习模型的训练方法,其特征在于,

所述密文计算结果为标量。

4.根据权利要求1～3中任一项所述的纵向联邦学习模型的训练方法,其特征在于,所述第一训练数据包括训练样本特征数据;

从本地获取所述明文算子包含的变量对应的第一训练数据,并代入所述明文算子得到明文计算结果,包括:

将所述训练样本特征数据代入所述明文算子得到明文计算结果;

将所述明文计算结果与密文计算结果代入所述任务算法得到数据对象,包括:

将所述明文计算结果与密文计算结果代入所述任务算法得到对应于所述训练样本特征数据的多个特征数据碎片;

将所述多个特征数据碎片中的全部或者部分发送至所述其他参与方,以由所述其他参与方根据其所拥有的特征数据碎片进行联邦学习模型的训练。

5.根据权利要求1～3中所述的纵向联邦学习模型的训练方法,其特征在于,所述第一训练数据包括训练样本特征数据的标签;

从本地获取所述明文算子包含的变量对应的第一训练数据,并代入所述明文算子得到明文计算结果,包括:

将所述标签代入所述明文算子得到明文计算结果;

将所述明文计算结果与密文计算结果代入所述任务算法得到数据对象,包括:

将所述明文计算结果与密文计算结果代入所述任务算法得到对应于所述训练样本特征数据的多个标签碎片;

将所述多个标签碎片中的全部或者部分发送至所述其他参与方,以由所述其他参与方根据其所拥有的标签碎片进行联邦学习模型的训练。

6.根据权利要求1～3中所述的纵向联邦学习模型的训练方法,其特征在于,所述第一训练数据包括本轮迭代所述联邦学习模型的输出结果;

从本地获取所述明文算子包含的变量对应的第一训练数据,并代入所述明文算子得到

明文计算结果，包括：

将所述输出结果代入所述明文算子得到明文计算结果；

将所述明文计算结果与密文计算结果代入所述任务算法得到数据对象，包括：

将所述明文计算结果与密文计算结果代入所述任务算法得到本轮迭代的多个梯度值碎片；

将所述多个梯度值碎片中的全部或者部分发送至所述其他参与方，以由所述其他参与方根据其所拥有的梯度值碎片进行联邦学习模型的训练。

7.根据权利要求1所述的纵向联邦学习模型的训练方法，其特征在于，所述训练方法用于训练逻辑回归模型；

训练所述逻辑回归模型中使用到的非线性变换函数如下：

$$f(x)=\frac{e^{w_a\cdot x_a}\cdot e^{w_b\cdot x_b}}{1+e^{w_a\cdot x_a}\cdot e^{w_b\cdot x_b}}$$。

8.根据权利要求1所述的纵向联邦学习模型的训练方法，其特征在于，从本地获取所述明文算子包含的变量对应的第一训练数据，并代入所述明文算子得到第一明文计算结果，包括：

对所述第一训练数据进行分组，将每组第一训练数据的均值代入所述明文算子得到第一明文计算结果；

所述训练方法还包括：第一训练数据的分组结果与所述第一训练数据的第一对应关系发送给其他参与方，以由其他参与方根据所述对应关系训练联邦学习模型。

9.根据权利要求1所述的纵向联邦学习模型的训练方法，其特征在于，所述第二明文计算结果由所述其他参与方对从其本地获取明文算子包含的变量对应的第二训练数据进行分组，并将每组第二训练数据的均值代入所述明文算子计算得到；

将所述第一明文计算结果与密文计算结果代入所述联邦算子进行安全计算，以得到用于联邦学习模型的训练的数据对象，包括：

根据第二对应关系，确定与各密文计算结果对应的第一明文计算结果，并代入所述联邦算子进行安全计算，以得到用于联邦学习模型的训练的数据对象；其中，所述第二对应关系由其他参与方提供，且所述第二对应关系表征所述第二训练数据的分组结果与所述第一训练数据的对应关系。

10.一种纵向联邦学习模型的训练装置，其特征在于，应用于多个参与方中的任一参与方，所述训练装置包括：

确定模块，用于响应于联邦学习任务，确定所述联邦学习任务涉及的任务算法，所述任务算法包括明文算子和联邦算子；

第一计算模块，用于从本地获取所述明文算子包含的变量对应的第一训练数据，并代入所述明文算子得到第一明文计算结果；

第二计算模块，用于将所述第一明文计算结果与密文计算结果代入所述联邦算子进行安全计算，以得到用于联邦学习模型的训练的数据对象；其中，所述密文计算结果为其他参与方提供的第二明文计算结果的密文形态，所述第二明文计算结果由所述其他参与方从其本地获取明文算子包含的变量对应的第二训练数据代入所述明文算子计算得到。

11.一种电子设备，包括存储器、处理器及存储在存储器上并可在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权利要求1至9任一项所述的纵向联邦学习模型的训练方法。

12.一种计算机可读存储介质，其上存储有计算机程序，其特征在于，所述计算机程序被处理器执行时实现权利要求1至9任一项所述的纵向联邦学习模型的训练方法。

纵向联邦学习模型的训练方法、装置、电子设备、介质

技术领域

[0001] 本发明涉及模型训练技术领域，尤其涉及一种纵向联邦学习模型的训练方法、装置、电子设备、介质。

背景技术

[0002] 联邦学习作为一种数据安全计算的技术，其能够实现在原始数据不出门的前提下，让数据价值在各个机构之间进行流动，创造业务价值，比如应用在金融风控、广告推荐等领域。联邦学习是一种分布式计算架构，支持多方安全计算，根据不同的业务使用场景，主要包括纵向联邦学习、横向联邦学习以及联邦迁移算法三种类型。目前联邦学习已经可以支持多种机器学习算法。

[0003] 目前对于纵向联邦学习，实现方式有很多，比如基于半同态加密、差分隐私、安全多方计算(MPC)等隐私保护手段。然而目前实现的各类方式，都伴随很多问题，特别是在高安全性、高精度、大规模计算场景。比如半同态加密算法，由于加密以及加密后的数值计算耗时，在大规模数据场景下，很难高效执行，并且基于半同态加密，模型其实也是暴露了一些信息，比如标签方相对特征方会拥有更多信息。而差分隐私通过牺牲精度的方式来提升计算性能，精度与模型识别准确率相关，如何找到平衡点也是难题。而采用多方安全计算(MPC或SMPC)，在安全性上具备较好的优势，且能够保证精度，并且计算效率高，但令人诟病的是多方安全计算的通信开销较大。

发明内容

[0004] 本发明要解决的技术问题是为了克服现有技术中纵向联邦学习模型的训练过程中通信开销很大的缺陷，提供一种纵向联邦学习模型的训练方法、装置、电子设备、介质。

[0005] 本发明是通过下述技术方案来解决上述技术问题：

[0006] 第一方面，提供一种纵向联邦学习模型的训练方法，应用于多个参与方中的任一参与方，所述训练方法包括：

[0007] 响应于联邦学习任务，确定所述联邦学习任务涉及的任务算法，所述任务算法包括明文算子和联邦算子；

[0008] 从本地获取所述明文算子包含的变量对应的第一训练数据，并代入所述明文算子得到第一明文计算结果；

[0009] 将所述第一明文计算结果与密文计算结果代入所述联邦算子进行安全计算，以得到用于联邦学习模型的训练的数据对象；其中，所述密文计算结果为其他参与方提供的第二明文计算结果的密文形态，所述第二明文计算结果由所述其他参与方从其本地获取明文算子包含的变量对应的第二训练数据代入所述明文算子计算得到。

[0010] 可选地，还包括：

[0011] 响应于数据获取请求，从本地获取所述明文算子包含的变量对应的第三训练数据，并代入所述明文算子得到第三明文计算结果；

[0012] 将所述第三明文计算结果的密文形态发送给所述其他参与方。

[0013] 可选地，

[0014] 所述密文计算结果为标量。

[0015] 可选地，所述第一训练数据包括训练样本特征数据；

[0016] 从本地获取所述明文算子包含的变量对应的第一训练数据，并代入所述明文算子得到明文计算结果，包括：

[0017] 将所述训练样本特征数据代入所述明文算子得到明文计算结果；

[0018] 将所述明文计算结果与密文计算结果代入所述任务算法得到数据对象，包括：

[0019] 将所述明文计算结果与密文计算结果代入所述任务算法得到对应于所述训练样本特征数据的多个特征数据碎片；

[0020] 将所述多个特征数据碎片中的全部或者部分发送至所述其他参与方，以由所述其他参与方根据其所拥有的特征数据碎片进行联邦学习模型的训练。

[0021] 可选地，所述第一训练数据包括训练样本特征数据的标签；

[0022] 从本地获取所述明文算子包含的变量对应的第一训练数据，并代入所述明文算子得到明文计算结果，包括：

[0023] 将所述标签代入所述明文算子得到明文计算结果；

[0024] 将所述明文计算结果与密文计算结果代入所述任务算法得到数据对象，包括：

[0025] 将所述明文计算结果与密文计算结果代入所述任务算法得到对应于所述训练样本特征数据的多个标签碎片；

[0026] 将所述多个标签碎片中的全部或者部分发送至所述其他参与方，以由所述其他参与方根据其所拥有的标签碎片进行联邦学习模型的训练。

[0027] 可选地，所述第一训练数据包括本轮迭代所述联邦学习模型的输出结果；

[0028] 从本地获取所述明文算子包含的变量对应的第一训练数据，并代入所述明文算子得到明文计算结果，包括：

[0029] 将所述输出结果代入所述明文算子得到明文计算结果；

[0030] 将所述明文计算结果与密文计算结果代入所述任务算法得到数据对象，包括：

[0031] 将所述明文计算结果与密文计算结果代入所述任务算法得到本轮迭代的多个梯度值碎片；

[0032] 将所述多个梯度值碎片中的全部或者部分发送至所述其他参与方，以由所述其他参与方根据其所拥有的梯度值碎片进行联邦学习模型的训练。

[0033] 可选地，所述训练方法用于训练逻辑回归模型；

[0034] 训练所述逻辑回归模型中使用到的非线性变换函数如下：

[0035] $$f(x)=\frac{e^{w_a\cdot x_a}\cdot e^{w_b\cdot x_b}}{1+e^{w_a\cdot x_a}\cdot e^{w_b\cdot x_b}}$$。

[0036] 可选地，从本地获取所述明文算子包含的变量对应的第一训练数据，并代入所述明文算子得到第一明文计算结果，包括：

[0037] 对所述第一训练数据进行分组，将每组第一训练数据的均值代入所述明文算子得到第一明文计算结果；

[0038] 所述训练方法还包括：第一训练数据的分组结果与所述第一训练数据的第一对应

关系发送给其他参与方，以由其他参与方根据所述对应关系训练联邦学习模型。

[0039] 可选地，所述第二明文计算结果由所述其他参与方对从其本地获取明文算子包含的变量对应的第二训练数据进行分组，并将每组第二训练数据的均值代入所述明文算子计算得到；

[0040] 将所述第一明文计算结果与密文计算结果代入所述联邦算子进行安全计算，以得到用于联邦学习模型的训练的数据对象，包括：

[0041] 根据第二对应关系，确定与各密文计算结果对应的第一明文计算结果，并代入所述联邦算子进行安全计算，以得到用于联邦学习模型的训练的数据对象；其中，所述第二对应关系由其他参与方提供，且所述第二对应关系表征所述第二训练数据的分组结果与所述第一训练数据的对应关系。

[0042] 第二方面，提供一种纵向联邦学习模型的训练装置，应用于多个参与方中的任一参与方，所述训练装置包括：

[0043] 确定模块，用于响应于联邦学习任务，确定所述联邦学习任务涉及的任务算法，所述任务算法包括明文算子和联邦算子；

[0044] 第一计算模块，用于从本地获取所述明文算子包含的变量对应的第一训练数据，并代入所述明文算子得到第一明文计算结果；

[0045] 第二计算模块，用于将所述第一明文计算结果与密文计算结果代入所述联邦算子进行安全计算，以得到用于联邦学习模型的训练的数据对象；其中，所述密文计算结果为其他参与方提供的第二明文计算结果的密文形态，所述第二明文计算结果由所述其他参与方从其本地获取明文算子包含的变量对应的第二训练数据代入所述明文算子计算得到。

[0046] 第三方面，提供一种电子设备，包括存储器、处理器及存储在存储器上并可在处理器上运行的计算机程序，所述处理器执行所述计算机程序时实现上述任一项所述的纵向联邦学习模型的训练方法。

[0047] 第四方面，提供一种计算机可读存储介质，其上存储有计算机程序，所述计算机程序被处理器执行时实现上述任一项所述的纵向联邦学习模型的训练方法。

[0048] 本发明的积极进步效果在于：本发明在联邦学习模型的训练过程中，数据计算包括明文计算和密文计算，也即实现了基于多态数据混合计算的联邦学习模型的训练优化，由于一部分数据在本地计算，无需通过密文的方式进行分享，因此能够极大降低通信量与通信次数，提高模型训练的效率，有效解决了在高安全性、低带宽、高延迟、大数据处理环境下，高效执行隐私保护的联邦学习模型的训练任务。对于低带宽、高延迟网络环境，更具有实用性和可行性。

附图说明

[0049] 图1为本发明一示例性实施例提供的一种纵向联邦学习模型的训练框架的结构示意图；

[0050] 图2为现有技术中的一种纵向联邦学习模型的训练方法的流程图；

[0051] 图3a为本发明一示例性实施例提供的一种纵向联邦学习模型的训练方法的流程图；

[0052] 图3b为本发明一示例性实施例提供的一种纵向联邦学习模型的训练过程中采用

的梯度碎片化的流程图；

[0053] 图4为本发明一示例性实施例提供的另一种纵向联邦学习模型的训练方法的流程图；

[0054] 图5a为本发明一示例性实施例提供的一种减法MPC计算逻辑；

[0055] 图5b为本发明一示例性实施例提供的一种减法MPC计算逻辑；

[0056] 图5c为本发明一示例性实施例提供的一种求最大值MPC计算逻辑；

[0057] 图6为本发明一示例性实施例提供的一种电子设备的结构示意图。

具体实施方式

[0058] 下面通过实施例的方式进一步说明本发明，但并不因此将本发明限制在所述的实施例范围之中。

[0059] 图1为本发明一示例性实施例提供的一种纵向联邦学习模型的训练框架的结构示意图。联邦学习中，参与方主要承担的角色有协调方、数据方和结果方。数据方是指提供联邦学习模型训练所需的私有数据的参与方。协调方是协调各参与方协作训练并使用联邦学习模型的参与方。结果方是指获取联邦学习结果的参与方。一个参与方可承担多个角色，例如一个参与方可以同时承担协调方、数据方和结果方三类角色。

[0060] 图2为现有技术中的一种纵向联邦学习模型的训练方法的流程图，图中采用LR-MPC算法(基于多方安全计算MPC协议的逻辑回归算法)中，模型训练发起方Guest既作为数据方(提供训练样本包含的标签以及部分特征)、也作为发起方(协调方)。特征方Host作为数据方，也作为计算参与方。图2示出的纵向联邦学习模型的训练过程中，数据均为密文态，存在以下几个缺点：

[0061] (1)全碎片化计算模式

[0062] 当前LR-MPC为端到端全碎片化计算模式。从整个生命周期来看，由数据输入-模型训练-模型验证-模型评估，全部为碎片化状态数值的MPC算子执行流程。

[0063] (2)当前LR-MPC中算子通信存在两个主要问题

[0064] A.通信量大:比如在Concat算子，会将全部训练数据的碎片进行向量分发。如果在500万样本量，80特征的场景，分发的量级非常庞大，以int64为例，大小在3G。乘法和矩阵乘法计算过程中的中间数据传输，同样也是向量，与输入的特征矩阵一致，同样非常庞大，也是以G为单位的量级。

[0065] B.通信次数多：

[0066] 综上，在这样的学习框架下，各方数据从一开始就被碎片化后秘密共享，整个执行流程只有密文碎片态形式的计算。虽然具备非常强的安全性，但由于多方安全计算秘密共享机制下的密文碎片态存在通信次数多，且分发的通信量大的问题，整体模型训练流程在低带宽、高延迟、大规模数据处理场景，非常耗时。

[0067] 基于上述问题，本发明实施例提供了一种纵向联邦学习模型的训练方法，图3a为本发明一示例性实施例提供的一种纵向联邦学习模型的训练方法的流程图，该训练方法应用于多个参与方中的任一参与方，该训练方法包括：

[0068] 步骤301、响应于联邦学习任务，确定联邦学习任务涉及的任务算法。

[0069] 其中，任务算法包括明文算子和联邦算子。

[0070] 联邦学习任务涉及的任务算法包括加、减、乘、除、矩阵乘法、求最大值、求均值、Sigmoid函数、Less than函数、Concat函数、Slice函数、Transpose函数等算子中的至少两项。根据各算子涉及的变量，将任务算子分为明文算子和联邦算子。明文算子用于对本参与方本地的数据进行计算，计算得到的第一明文计算结果不发送给其他参与方；联邦算子对密文数据或者半密文数据进行计算，也即联邦算子包含的变量全部为密文数据或者部分为密文数据，例如根据本参与方计算得到的第一明文计算结果(明文数据或者密文数据)和其他参与方发送的密文计算结果(密文数据)计算数据对象。

[0071] 通过将多方安全MPC算法分为明文算子和联邦算子两部分，可以最小化MPC算法的计算任务，无需联合交互执行的任务尽可能在本地计算完成，且通信尽可能采用标量形式，而非向量形式，能够极大降低通信量与通信次数，减少数据通信的功耗。

[0072] 步骤302、从本地获取明文算子包含的变量对应的第一训练数据，并代入明文算子得到第一明文计算结果。

[0073] 其中，第一训练数据为联邦学习模型训练所需的数据，可以包括联邦学习模型训练过程中的样本数据和/或中间过程数据。中间过程数据包括联邦学习模型每次迭代的输出结果。可以理解的，对于模型训练的不同阶段，中间过程数据包含的参数可能不同，举例来说，损失函数计算涉及的变量参数与梯度计算涉及的变量参数不同。

[0074] 在一个实例中，可以在参与方中设置调度层，由调度层判断任务算法中哪部分可以明文计算，哪部分进行联邦计算。在一种实现方式中，可以通过判断算法包含的变量是否能够在本地查找到相应的数据，来确定任务算法中的明文计算部分、联邦计算部分。

[0075] 下面介绍步骤302的居然具体实现方式。

[0076] 以任务算法为(D1+3)×(D2+5)为例，D1和D2分别表示所属于A和B两家公司的两份数据，这两份数据都不能泄露给对方。假设D1+3的计算结果为DA，D2+5的计算结果为DB，DA×DB的结果为DC。

[0077] 因为(D1+3)和(D2+5)在计算时不具有前后依赖关系，因此可以同步进行，这里将(D1+3)调度到A公司的参与方进行本地计算，对A公司作为参与方来说，(D1+3)属于明文算子；将(D2+5)调度到B公司作为参与方进行本地计算，对B公司作为参与方来说，(D2+5)属于明文算子。而DA×DB涉及双方数据，因此其为联邦算法。

[0078] A公司作为参与方进行策略判断：D1属于本地数据，+3的操作属于单节点内部操作，并且明文算子D1+3不会造成数据泄露，因此采用本地明文进行计算，计算结果D1+3为明文数据。D2不属于本地数据，发送指令给B公司参与方，要求获取DB的密文形态。

[0079] B公司作为参与方进行策略判断：D2属于本地数据，+5的操作属于单节点内部操作，并且明文算子D2+5不会造成数据泄露，因此采用本地明文进行计算，计算结果D2+5为明文数据。D1不属于本地数据，发送指令给A公司参与方，要求获取DA的密文形态。

[0080] 对于DA×DB，因为DA是A公司计算后得到的数据，DB是B公司数据计算后得到的数据，为了避免数据泄露，这步操作需要密文情况下进行，DA×DB为联邦算子。A公司作为参与方对DA进行加密处理，获取到DA数据的密文形态。DB数据因为存在于远程B公司，所以这里就需要2个公司密文节点的协商通信过程。这时候公司A作为参与方发现DB是远程节点的数据，则发送一个指令给B公司，要求获取DB数据的密文形态，B公司作为参与方找到生成的DB数据，并获取DB的密文形态，返回给A公司参与方，这时A公司参与方就获取到了DB的密文形

态,可以进行DA×DB计算,并且生成结果DC,DC是密文。在多方安全计算中,采用的就是DA数据和DB数据在2个密文节点通过秘密共享方式来进行安全计算。

[0081] 上述计算过程,即包括密文计算,也包括明文计算,是一种基于多态数据混合计算逻辑。在开发多态数据混合计算逻辑时,可以在同一套代码中实现明文计算和密文计算,在上层增加调度策略层。参与方的控制端把联邦学习任务拆分成n个子任务以及并发执行子任务的调度完成任务算法的n个算子,该n个算子中的部分为明文算子,另外一部分为联邦算子。

[0082] 控制端会基于数据的状态类型以及任务类型进行判断,比如任务为单方节点的两个数据计算,则会使用对应单方节点的计算逻辑;如果任务涉及到多方协同计算或者计算对象是密文态,则会转换为密文态或者在密文态下进行相应联邦计算。控制端不需要区别的判断某一次计算应该是明文还是密文的,也不需要进行显式的进行数据的加密解密操作,这是由于任务会自动判断任务类型或者数据形态,因此不需要显式在程序代码上面显式把代码进行划分成明文代码或者密文代码。由系统计算判断是否需要进行数据形态的转换以及计算算子的调度。在任务分为成子任务后,不用区别的对待计算节点,进行统一的分布式调度即可,是一种无感知的混合计算的联邦学习模型的训练框架。

[0083] 需要说明的是,上述只是以A公司和B公司两个参与方进行举例说明,实际应用过程中,参与方的数量不限于两个,还可以是三个、四个、甚至更多。

[0084] 步骤303、将第一明文计算结果与密文计算结果代入联邦算子进行安全计算,以得到用于联邦学习模型的训练数据对象。

[0085] 其中,密文计算结果为其他参与方提供的第二明文计算结果的密文形态,第二明文计算结果由其他参与方从其本地获取明文算子包含的变量对应的第二训练数据代入明文算子计算得到。

[0086] 还是以任务算法为(D1+3)×(D2+5)为例,对于A公司参与方来说,(D1+3)为第一明文计算结果,(D2+5)的密文形态为B公司参与方(其他参与方)提供的密文计算结果。

[0087] 在一个实施例中,直接将第一明文计算结果和密文计算结果代入联邦算子进行安全计算,得到训练数据对象。

[0088] 在一个实施例中,将第一明文计算结果转化为密文形态,并将第一明文计算结果的密文形态和密文计算结果代入联邦算子进行安全计算,得到训练数据对象。

[0089] 上述密文形态的转化,可以采用现有技术提供的加密算法,此处不再赘述。

[0090] 在一个实施例中,得到数据对象后,在数据对象上进行一次封装,将数据的不同形态(包括但不限于明文形态、密文形态)打包成一个完整的数据,这个完整的数据命名为数据DS。这个数据DS自己提供数据形态转换的接口。

[0091] 从而,当用户需要使用数据的不同形态时,不用显式的去调用加解密方法来转换数据形态,只需要调用数据DS的get_public_data或者get_private_data就可以获取到自己需要的数据。对于初始输入的私有数据,比如A公司有数据D1,B公司有数据D2,这些初始私有数据,在计算开始的时候,创建对应的数据DS对象,如果是明文私有态的数据,通过set_local_data将D1(B公司就是D2)设置到对应的数据DS对象中;如果是明文共享态,则通过set_public_data由发起方广播给各参与方;如果是密文的数据,通过set_private_data将D1(B公司就是D2)设置到对应的数据DS对象中。

[0092] 也就是说,各参与方可以通过get_public_data、get_private_data、set_local_data和set_public_data这些指令的调用,即可获取到其想要或者需要的数据。通过上述指令,使得混合运算编程对于开发者可以做到无感知,在底层混合运算算子开发中,可以对数据进行多态转换,在多态上层做了数据的统一封装,由参与方的计算系统内部进行明文或者密文计算的选择适配,可以对编程开发人员进行无门槛转换,降低开发难度。可有效支撑混合计算算子的开发,支撑混合计算框架的落地。

[0093] 关于明文或者密文计算的选择适配,在对应数据形态算子计算过程中,会自动识别数据类型以及数据所属节点一致性判别,如果执行的是local计算,也就是当前的数据类型为LocalTensor或者涉及的计算数据所属节点为同一个节点,则会进行local形态进行计算;在执行密态计算,也就是当前数据类型为PrivateTensor或者涉及的计算数据所属节点不同,则会转换为private类型进行计算。在发明实施例提供的混合运算框架中,编程用户无需关心在什么节点采用何种形态,只需要调用相应的混合运算算子即可,数据形态做了上层的统一封装,暴露给用户的为统一的DS对象。

[0094] 本发明实施例中,在联邦学习模型的训练过程中,数据计算包括明文计算和密文计算,也即实现了基于多态数据混合计算的联邦学习模型的训练优化,由于一部分数据在本地计算,无需通过密文的方式进行分享,因此能够极大降低通信量与通信次数,提高模型训练的效率,有效解决了在高安全性、低带宽、高延迟、大数据处理环境下,高效执行隐私保护的联邦学习模型的训练任务。对于低带宽、高延迟网络环境,更具有实用性和可行性。

[0095] 在一个实施例中,第一训练数据包括训练样本特征数据;步骤302中,从本地获取明文算子包含的变量对应的第一训练数据,并代入明文算子得到明文计算结果包括:将训练样本特征数据代入明文算子得到明文计算结果。步骤303中,将明文计算结果与密文计算结果代入任务算法得到数据对象包括:将明文计算结果与密文计算结果代入任务算法得到对应于训练样本特征数据的多个特征数据碎片;将多个特征数据碎片中的全部或者部分发送至其他参与方,以由其他参与方根据其所拥有的特征数据碎片进行联邦学习模型的训练。

[0096] 特征数据碎片为训练样本特征数据的密文碎片态,特征数据碎片为对训练样本特征数据进行碎片化处理得到,碎片化处理是指基于多方安全计算秘密共享机制,对数据进行分片,各参与方只能持有部分碎片信息,具体持有多少的碎片信息,由多方安全协议规范所决定。比如在SPDZ或者NPZ等多方安全计算协议中,假如有3个计算参与方,数据X分拆成(X1,X2,X3),一个参与方只能持有其中一份碎片,并且各方不重复,即A方持有X1,B方持有X2,C方持有X3,只有当三方的数据合并在一起,才能恢复出原始数据X。又比如在aby3多方安全计算协议中,同样假设有3个计算参与方,数据X分拆(X1,X2,X3),一个参与方持有其中两份碎片,即A方持有(X1,X2),B方持有(X2,X3),C方持有(X3,X1),任意两方的数据合并在一起,可以恢复出原始数据X。这里的X1,X2,X3就被称为是原始数据X的密文碎片态。明文私有态是指数据X在本地阶段节点独有,不会出本地,没有被分发到其他参与节点。明文共享态则是指数据X在各个节点被明文共享,也就是A方有X,B方有X,C方也有X。

[0097] 通过对各参与方的训练样本特征数据进行秘密分享,使得在不泄漏其他参与方的数据的前提下,各参与方获得模型训练所需的完整的训练样本特征数据。举例来说,训练联邦学习模型所需的训练样本特征包括特征a、特征b、特征c和特征d,但是A公司参与方只有

特征a和特征b，缺少特征c和特征d，通过秘密分享使得A公司参与方得到特征a、特征b、特征c和特征d，但是A公司参与方又无法得知特征c和特征d的具体数值。

[0098] 在一个实施例中，第一训练数据包括训练样本特征数据的标签；步骤302中，从本地获取明文算子包含的变量对应的第一训练数据，并代入明文算子得到明文计算结果包括：将标签代入明文算子得到明文计算结果；步骤303中，将明文计算结果与密文计算结果代入任务算法得到数据对象包括：将明文计算结果与密文计算结果代入任务算法得到对应于训练样本特征数据的多个标签碎片；将多个标签碎片中的全部或者部分发送至其他参与方，以由其他参与方根据其所拥有的标签碎片进行联邦学习模型的训练。

[0099] 在一个实施例中，第一训练数据包括本轮迭代联邦学习模型的输出结果；步骤302中，从本地获取明文算子包含的变量对应的第一训练数据，并代入明文算子得到明文计算结果包括：将输出结果代入明文算子得到明文计算结果；步骤303中，将明文计算结果与密文计算结果代入任务算法得到数据对象包括：将明文计算结果与密文计算结果代入任务算法得到本轮迭代的多个梯度值碎片；将多个梯度值碎片中的全部或者部分发送至其他参与方，以由其他参与方根据其所拥有的梯度值碎片进行联邦学习模型的训练。

[0100] 应用于低带宽、高延迟、高精度、大规模数据处理的纵向联邦学习训练场景。这里所提的多态，是指数据存在密文碎片态、明文私有态、明文共享态等。

[0101] 本发明实施例中，实现了一种无感知的多态动态转换模式，在对应数据形态算子计算过程中，会自动识别数据类型，如果执行的是local计算，则会进行local形态转换；在执行密态计算，则会转换为private类型，加密方法在本文中为多方安全计算中的碎片化秘密共享模式。在本文所示混合运算框架中，编程用户无需关心在什么节点采用何种形态，只需要调用相应的混合运算算子即可，数据形态做了上层的统一封装，暴露给用户的为统一的DS对象。

[0102] 在一个实施例中，在进行计算之前，先对第一训练数据进行分组，然后将每组第一训练数据的均值代入明文算子得到第一明文计算结果。先分组再计算，能够极大地减少计算量。

[0103] 其中，对第一训练数据进行分组的方式可以但不限于包括聚类、直方图处理。

[0104] 参与方还将第一训练数据的分组结果与第一训练数据的第一对应关系发送给其他参与方，以由其他参与方根据对应关系训练联邦学习模型。

[0105] 在一个实施例中，第二明文计算结果由其他参与方对从其本地获取明文算子包含的变量对应的第二训练数据进行分组，并将每组第二训练数据的均值代入明文算子计算得到。步骤303包括：根据第二对应关系，确定与各密文计算结果对应的第一明文计算结果，并代入联邦算子进行安全计算，以得到用于联邦学习模型的训练的数据对象；其中，第二对应关系由其他参与方提供，且第二对应关系表征第二训练数据的分组结果与第一训练数据的对应关系。

[0106] 本发明实施例中，通过对训练数据进行分组能够大大减少计算量，提高模型训练的效率。

[0107] 参见图3b，以基于MPC算法对梯度进行碎片化为例，图中参与方Host将第一训练数据进行分组，也即将包含M个用户且每个用户包含N个特征参数x_{ij}的第一训练数据进行分组，得到K个分组，其中K远小于M。图中分别对特征参数salary和gender进行分组为例，分别

将两个特征进行分组(或者分箱)。将每个分组内的特征参数的均值作为该分组的代表,得到表征第一训练数据的新矩阵K×N,该新矩阵K×N的数据量远小于表征第一训练数据的原矩阵M×N,因此能够大大减少计算量,提高模型训练的效率;且经过试验可知,采用新矩阵K×N进行安全计算并不会影响训练结果的精确度。其中,图中的梯度计算公式中x_{ij}表示第i个用户的第j个特征,Y_i'表示本轮迭代模型的输出结果,Y_i表示训练标签,z_i表示正则项。

[0108] 参与方Host还将分组的对应关系(索引)发送给参与方Gust,对应关系表征每个分组包含的哪些用户的特征参数数据,以由参与方Gust根据该对应关系从本地匹配对应的数据,计算碎片化梯度。

[0109] 本发明实施例提供的纵向联邦学习模型的训练方法可以用于训练各类模型,例如金融领域的风控预测模型、信息推荐模型等;医疗领域的辅助诊断模型、医学影像识别模型等;数据分析领域的分类模型或者逻辑回归模型;当然还包括其他领域,此处不再追溯。

[0110] 训练得到的风控预测模型,可以用于风控预测。训练得到的信息推荐模型,可以向用户推荐其所需的、个性化的信息。训练得到的辅助诊断模型,可以基于医学图像数据或者其他医学数据辅助医护人员进行医疗诊断。训练得到的医学影像识别模型,可以基于医学图像数据进行医学影像识别(分割)。训练得到的逻辑回归模型可以解决分类问题。

[0111] 参见图4,图中以在纵向联邦学习的场景下完成逻辑回归模型的训练,以得到银行与运营商之间联合建模评分卡模型为例,对模型训练过程作进一步说明,图中GUEST作为模型训练发起方,HOST作为数据提供方。

[0112] 图4涉及的任务算法包括以下至少一种sigmoid算法、loss算法、梯度算法、Early stop算法。

[0113] 图4中关键节点的数据通信交互,不再是向量形式,而是采用标量形式。比如在图2中,对特征向量x是直接进行碎片化,然后将其中的一份碎片通信发送给另一方。假如x是500万行,1000列的矩阵,这个数据是非常庞大的,以int64类型计算,达到了将近38GB。这在低带宽、高延迟的网络环境下,是非常耗时,甚至是糟糕的。且在逻辑回归中使用到的非线性变换函数sigmoid,标准的公式如下:

[0114] $$f(u)=\frac{1}{1+e^{-u}};$$

[0115] 由于MPC没法直接进行非线性计算,必须转换为多项式计算,也即下述分段函数公式:

[0116] $$f(x)=\begin{cases}0, & if\ x<-0.5\\ x+0.5, & if\ -0.5\le x\le 0.5\\ 1, & if\ x>0.5\end{cases};$$

[0117] 然而这个公式每次执行需要执行2次MPC比较,以及其他的计算,以64bit大小的数值为例,整体通信次数达到了240次,计算量很大。

[0118] 在发明实施例中,对函数sigmoid进行了调整,结合混合运算思想,将函数sigmoid改写为如下所示:

[0119] $$f(x)=\frac{1}{1+e^{-(w_a\cdot x_a+w_b\cdot x_b)}}=\frac{e^{w_a\cdot x_a}\cdot e^{w_b\cdot x_b}}{1+e^{w_a\cdot x_a}\cdot e^{w_b\cdot x_b}};$$

[0120] 参见图4,本发明实施例中采用了标量的形式,也就是先在本地计算权重参数与特征的矩阵乘法,即q＝w@x,@符号表示的是矩阵乘法,还进一步计算了e^q,基于改写后的函数sigmoid,可以仅采用一次MPC乘法,即可实现计算,整体通信次数由240次压缩到8次。本发明实施例所描述的MPC算子通信次数都以NPZ协议为例,如果采用SPDZ协议,由于其采用了线下计算模式,在线计算可以压缩到4次通信。以此说明的是本发明实施例所提混合运算可以极大程度减少通信次数。

[0121] 图4中创新性地引入了混合运算sigmoid变形函数表达、混合运算梯度计算公式、混合运算Max算子,支撑模型的混合计算框架搭建,显著降低了通信量和通信次数,且尽可能利用本地计算完成数值的预处理。另外,通过引入混合计算算子,可以在某些关键步骤,做到多方安全计算算子执行,与数据量级无关,有效应对在大数据处理场景下耗时线性甚至非线性增加的难题。

[0122] 需要说明的是,本发明实施例提供纵向联邦学习的混合计算思路,不局限在联邦逻辑回归模型,还可以应用于联邦深度学习、联邦树模型学习等场景,通过降低通信量与通信次数,通过本地计算与多方安全计算的混合,极大提升了联邦学习的计算性能,在利用多方安全计算的高精度、高安全性特点的同时,又明显提升了隐私保护的联邦学习效率。

[0123] 进一步的,对于梯度的计算,也可以拆分成结合混合运算的模式,在本文中,对于正则项进行了本地预先计算,再结合MPC乘法,即可算出对应特征的梯度值,如图4所示,GUEST和HOST两方可以在MPC计算出碎片化梯度值之后,分别恢复各自特征所对应的梯度明文私有态的数值。而在计算过程中,比如可以看到损失值公式中的log2,这类数值是明文共享态,可以在各个参与方之间以明文形式存在,参与计算。公式中[]形式的数值,表示的是密文碎片态数值,采用MPC算子形式计算。这里给出一些示例以帮助理解,比如图5a,图5b分别展示的是减法和乘法的MPC计算逻辑。

[0124] 另外,本发明实施例采用的早停机制(Early Stop),是将最大梯度值与阈值进行相比,如果最大梯度值已经小于阈值,说明权重参数更新的幅度已经非常小了,模型处于相对稳定状态,也就是趋于收敛,这个时侯终止模型训练并进行模型保存。因此早停机制,涉及最大值和阈值比较两类计算。在图2的端到端框架下,最大值计算采用的是密文碎片态的计算,当特征数特别多的时侯,需要进行复杂度O(nlog(n))次比较,通信次数非常多。而采用混合计算的max算子,如图5c所示,只需要1次比较,复杂度是常数级O(1),可以做到与特征数无关,极大提升计算效率,粗略估计,可以减少几十倍的通信次数。

[0125] 因此,本发明实施例提供纵向混合运算框架中,参数更新可以在本地基于明文私有态数值计算完成,不需要进行MPC的通信交互。从整体流程来看,混合计算模式尽可能利用本地运算,全流程最小化MPC计算,此外通信数据尽可能使用标量,减少向量传输。

[0126] 粗略估算,以500w 1000f数据集训练为例,单batch,涉及的主要算子

[0127] 数及通信次数:

[0128]

任务算法	MPC 算子	显式调用次数	通信次数	优化数据量
sigmoid 计算	乘法(标量)	1	8	减少 1000 倍
loss 计算	乘法(标量)	2	16	/
梯度计算	乘法(标量)	2	16	/
Early stop	最大值(单值) 比较(单值)	2 次比较	212	实现与特征数无关
总和	/	/	252	

[0129] 从通信次数和通信量级评估,多态混合运算模型训练框架,可以降低几十到上百倍。在低带宽、高延迟、大数据处理的网络环境下,可以快速执行模型的训练迭代任务。

[0130] 与前述纵向联邦学习模型的训练方法实施例相对应,本发明还提供了纵向联邦学习模型的训练装置的实施例。

[0131] 对于装置实施例而言,由于其基本对应于方法实施例,所以相关之处参见方法实施例的部分说明即可。以上所描述的装置实施例仅仅是示意性的,其中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模块来实现本发明方案的目的。本领域普通技术人员在不付出创造性劳动的情况下,即可以理解并实施。

[0132] 图6为本发明一示例实施例示出的一种电子设备的结构示意图,示出了适于用来实现本发明实施方式的示例性电子设备60的框图。图6显示的电子设备60仅仅是一个示例,不应对本发明实施例的功能和使用范围带来任何限制。

[0133] 如图6所示,电子设备60可以以通用计算设备的形式表现,例如其可以为服务器设备。电子设备60的组件可以包括但不限于:上述至少一个处理器61、上述至少一个存储器62、连接不同系统组件(包括存储器62和处理器61)的总线63。

[0134] 总线63包括数据总线、地址总线和控制总线。

[0135] 存储器62可以包括易失性存储器,例如随机存取存储器(RAM)621和/或高速缓存存储器622,还可以进一步包括只读存储器(ROM)623。

[0136] 存储器62还可以包括具有一组(至少一个)程序模块624的程序工具625(或实用工具),这样的程序模块624包括但不限于:操作系统、一个或者多个应用程序、其它程序模块以及程序数据,这些示例中的每一个或某种组合中可能包括网络环境的实现。

[0137] 处理器61通过运行存储在存储器62中的计算机程序,从而执行各种功能应用以及数据处理,例如上述任一实施例所提供的方法。

[0138] 电子设备60也可以与一个或多个外部设备64(例如键盘、指向设备等)通信。这种通信可以通过输入/输出(I/O)接口65进行。并且,模型生成的电子设备60还可以通过网络适配器66与一个或者多个网络(例如局域网(LAN),广域网(WAN)和/或公共网络,例如因特

网)通信。如图所示,网络适配器66通过总线63与模型生成的电子设备60的其它模块通信。应当明白,尽管图中未示出,可以结合模型生成的电子设备60使用其它硬件和/或软件模块,包括但不限于:微代码、设备驱动器、冗余处理器、外部磁盘驱动阵列、RAID(磁盘阵列)系统、磁带驱动器以及数据备份存储系统等。

[0139] 应当注意,尽管在上文详细描述中提及了电子设备的若干单元/模块或子单元/模块,但是这种划分仅仅是示例性的并非强制性的。实际上,根据本发明的实施方式,上文描述的两个或更多单元/模块的特征和功能可以在一个单元/模块中具体化。反之,上文描述的一个单元/模块的特征和功能可以进一步划分为由多个单元/模块来具体化。

[0140] 本发明实施例还提供一种计算机可读存储介质,其上存储有计算机程序,所述程序被处理器执行时实现上述任一实施例所提供的方法。

[0141] 其中,可读存储介质可以采用的更具体可以包括但不限于:便携式盘、硬盘、随机存取存储器、只读存储器、可擦拭可编程只读存储器、光存储器件、磁存储器件或上述的任意合适的组合。

[0142] 在可能的实施方式中,本发明实施例还可以实现为一种程序产品的形式,其包括程序代码,当所述程序产品在终端设备上运行时,所述程序代码用于使所述终端设备执行实现上述任一实施例的方法。

[0143] 其中,可以以一种或多种程序设计语言的任意组合来编写用于执行本发明的程序代码,所述程序代码可以完全地在用户设备上执行、部分地在用户设备上执行、作为一个独立的软件包执行、部分在用户设备上部分在远程设备上执行或完全在远程设备上执行

[0144] 虽然以上描述了本发明的具体实施方式,但是本领域的技术人员应当理解,这仅是举例说明,本发明的保护范围是由所附权利要求书限定的。本领域的技术人员在不背离本发明的原理和实质的前提下,可以对这些实施方式做出多种变更或修改,但这些变更和修改均落入本发明的保护范围。

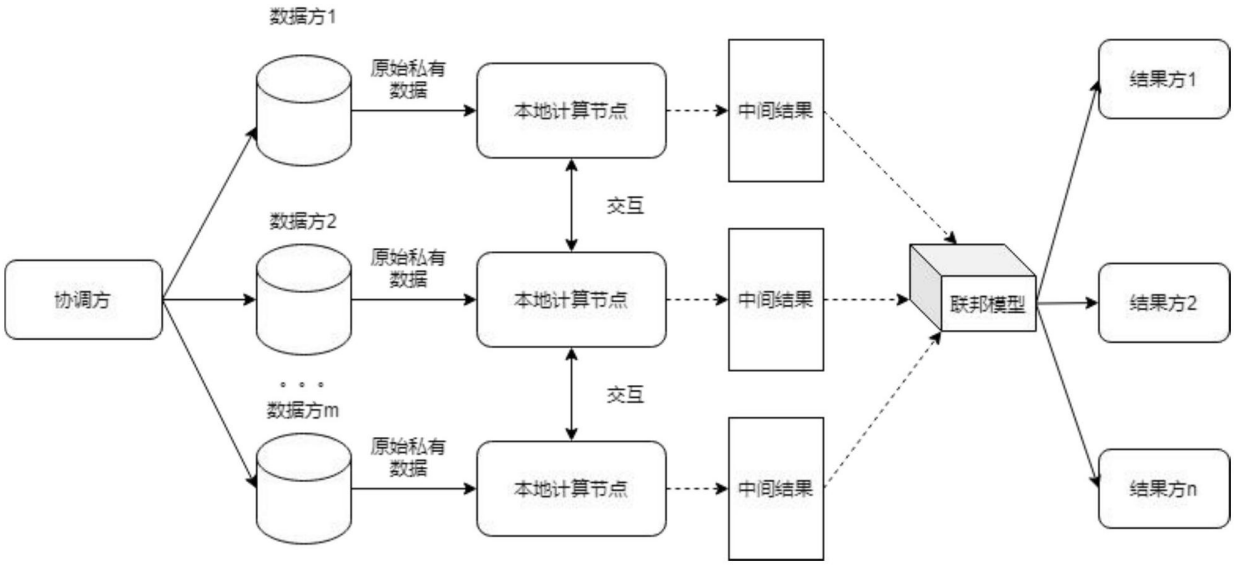

数据方1
原始私有数据
本地计算节点
中间结果
结果方1
交互
数据方2
协调方
原始私有数据
本地计算节点
中间结果
联邦模型
结果方2
交互
数据方m
原始私有数据
本地计算节点
中间结果
结果方n

图1

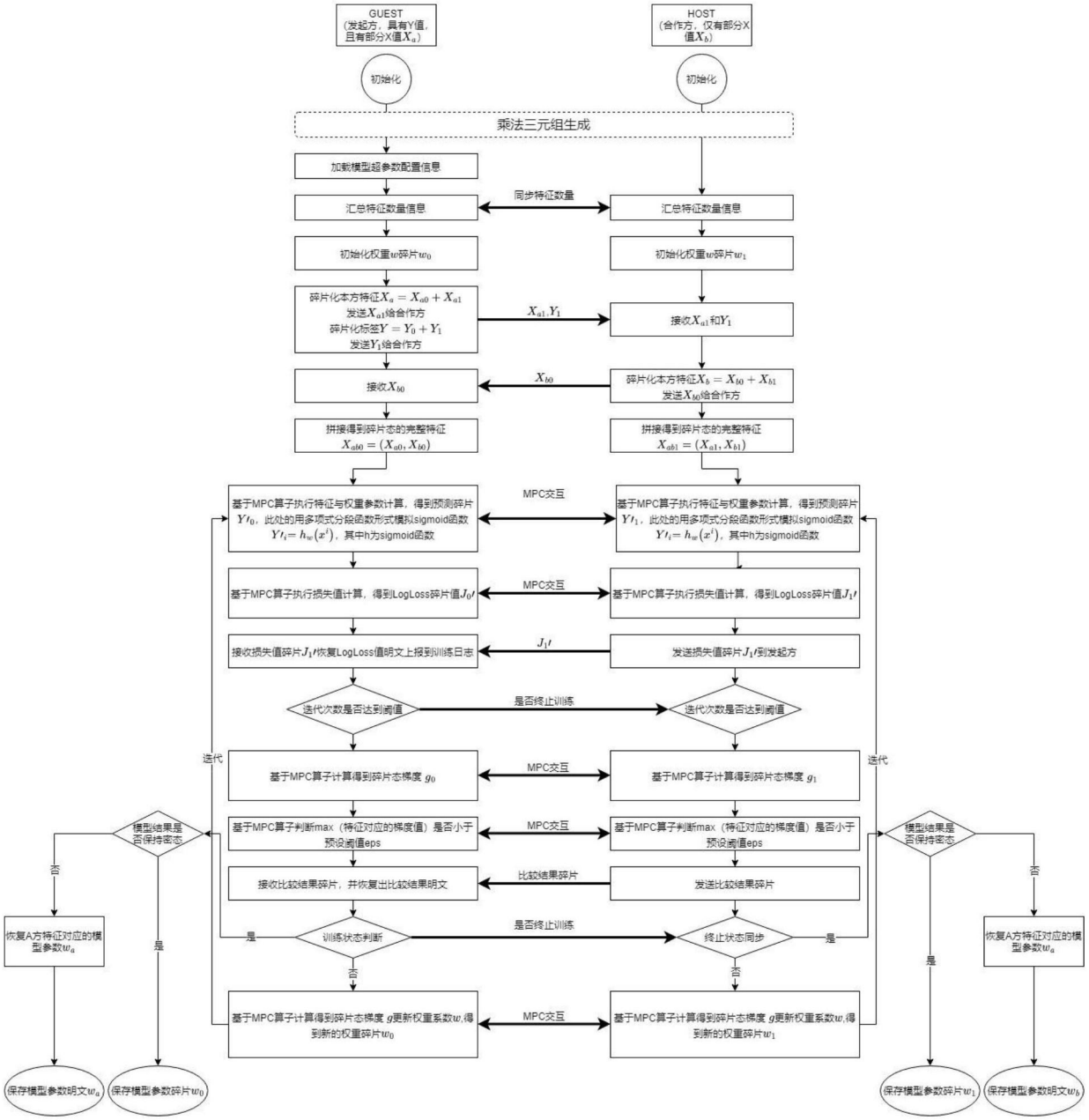

GUEST
(发起方，具有Y值，且有部分X值X_a)
HOST
(合作方，仅有部分X值X_b)
初始化
初始化
乘法三元组生成
加载模型超参数配置信息
同步特征数量
汇总特征数量信息
汇总特征数量信息
初始化权重w碎片w_0
初始化权重w碎片w_1
碎片化本方特征$X_a = X_{a0} + X_{a1}$
发送X_{a1}给合作方
碎片化标签$Y = Y_0 + Y_1$
发送Y_1给合作方
X_{a1}, Y_1
接收X_{a1}和Y_1
接收X_{b0}
X_{b0}
碎片化本方特征$X_b = X_{b0} + X_{b1}$
发送X_{b0}给合作方
拼接得到碎片态的完整特征
$X_{ab0} = (X_{a0}, X_{b0})$
拼接得到碎片态的完整特征
$X_{ab1} = (X_{a1}, X_{b1})$
基于MPC算子执行特征与权重参数计算，得到预测碎片Yt_0，此处的用多项式分段函数形式模拟sigmoid函数$Yt_i = h_w(x^i)$，其中h为sigmoid函数
MPC交互
基于MPC算子执行特征与权重参数计算，得到预测碎片Yt_1，此处的用多项式分段函数形式模拟sigmoid函数$Yt_i = h_w(x^i)$，其中h为sigmoid函数
基于MPC算子执行损失值计算，得到LogLoss碎片值J_0t
MPC交互
基于MPC算子执行损失值计算，得到LogLoss碎片值J_1t
接收损失值碎片J_1t恢复LogLoss值明文上报到训练日志
J_1t
发送损失值碎片J_1t到发起方
迭代次数是否达到阈值
是否终止训练
迭代次数是否达到阈值
迭代
基于MPC算子计算得到碎片态梯度 g_0
MPC交互
基于MPC算子计算得到碎片态梯度 g_1
迭代
模型结果是否保持密态
基于MPC算子判断max（特征对应的梯度值）是否小于预设阈值eps
MPC交互
基于MPC算子判断max（特征对应的梯度值）是否小于预设阈值eps
模型结果是否保持密态
否
接收比较结果碎片，并恢复出比较结果明文
比较结果碎片
发送比较结果碎片
否
恢复A方特征对应的模型参数w_a
是
是
训练状态判断
是否终止训练
终止状态同步
是
恢复A方特征对应的模型参数w_a
否
否
是
基于MPC算子计算得到碎片态梯度 g更新权重系数w,得到新的权重碎片w_0
MPC交互
基于MPC算子计算得到碎片态梯度 g更新权重系数w,得到新的权重碎片w_1
保存模型参数明文w_a
保存模型参数碎片w_0
保存模型参数碎片w_1
保存模型参数明文w_b

图2

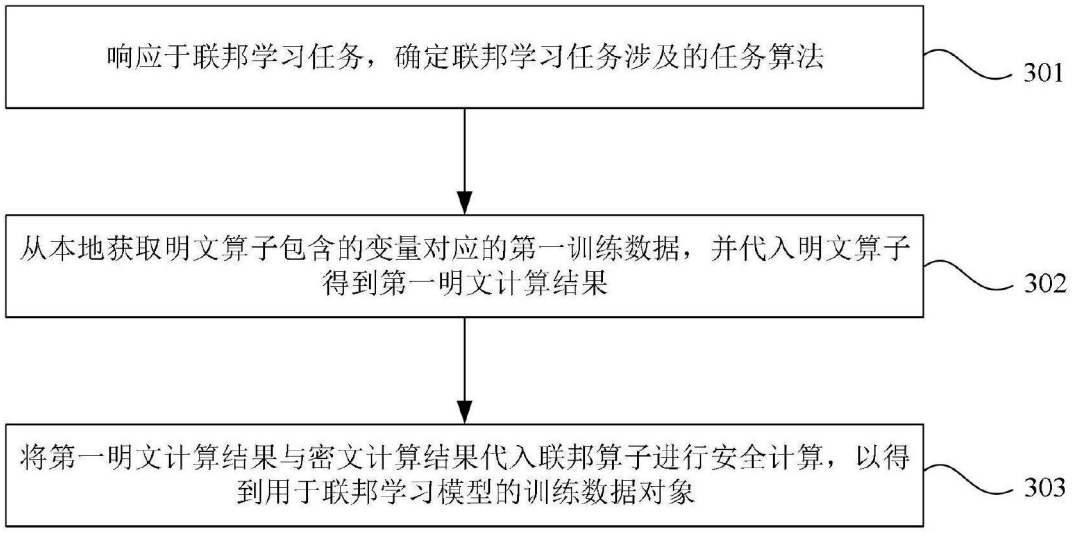

响应于联邦学习任务，确定联邦学习任务涉及的任务算法
301
从本地获取明文算子包含的变量对应的第一训练数据，并代入明文算子得到第一明文计算结果
302
将第一明文计算结果与密文计算结果代入联邦算子进行安全计算，以得到用于联邦学习模型的训练数据对象
303

图3a

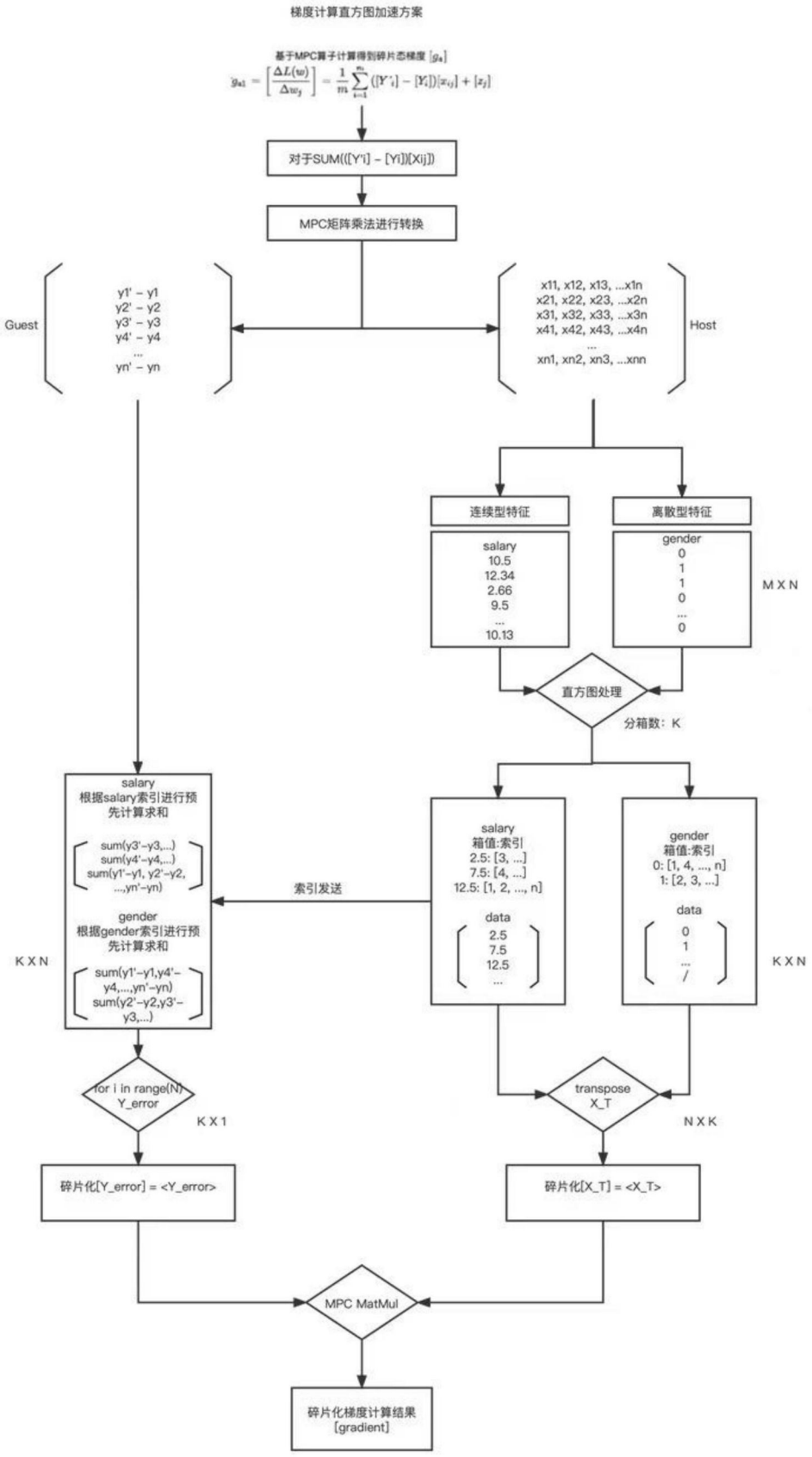

梯度计算直方图加速方案
基于MPC算子计算得到碎片态梯度 [g_a]
对于SUM(([Y'i] - [Yi])[Xij])
MPC矩阵乘法进行转换
Guest
y1' - y1
y2' - y2
y3' - y3
y4' - y4
...
yn' - yn
x11, x12, x13, ...x1n
x21, x22, x23, ...x2n
x31, x32, x33, ...x3n
x41, x42, x43, ...x4n
...
xn1, xn2, xn3, ...xnn
Host
连续型特征
离散型特征
salary
10.5
12.34
2.66
9.5
...
10.13
gender
0
1
1
0
...
0
M X N
直方图处理
分箱数：K
salary
根据salary索引进行预先计算求和
sum(y3'-y3,...)
sum(y4'-y4,...)
sum(y1'-y1, y2'-y2, ...,yn'-yn)
gender
根据gender索引进行预先计算求和
sum(y1'-y1,y4'-y4,...,yn'-yn)
sum(y2'-y2,y3'-y3,...)
K X N
索引发送
salary
箱值:索引
2.5: [3, ...]
7.5: [4, ...]
12.5: [1, 2, ..., n]
data
2.5
7.5
12.5
...
gender
箱值:索引
0: [1, 4, ..., n]
1: [2, 3, ...]
data
0
1
...
/
K X N
for i in range(N)
Y_error
K X 1
transpose
X_T
N X K
碎片化[Y_error] = <Y_error>
碎片化[X_T] = <X_T>
MPC MatMul
碎片化梯度计算结果
[gradient]

图3b

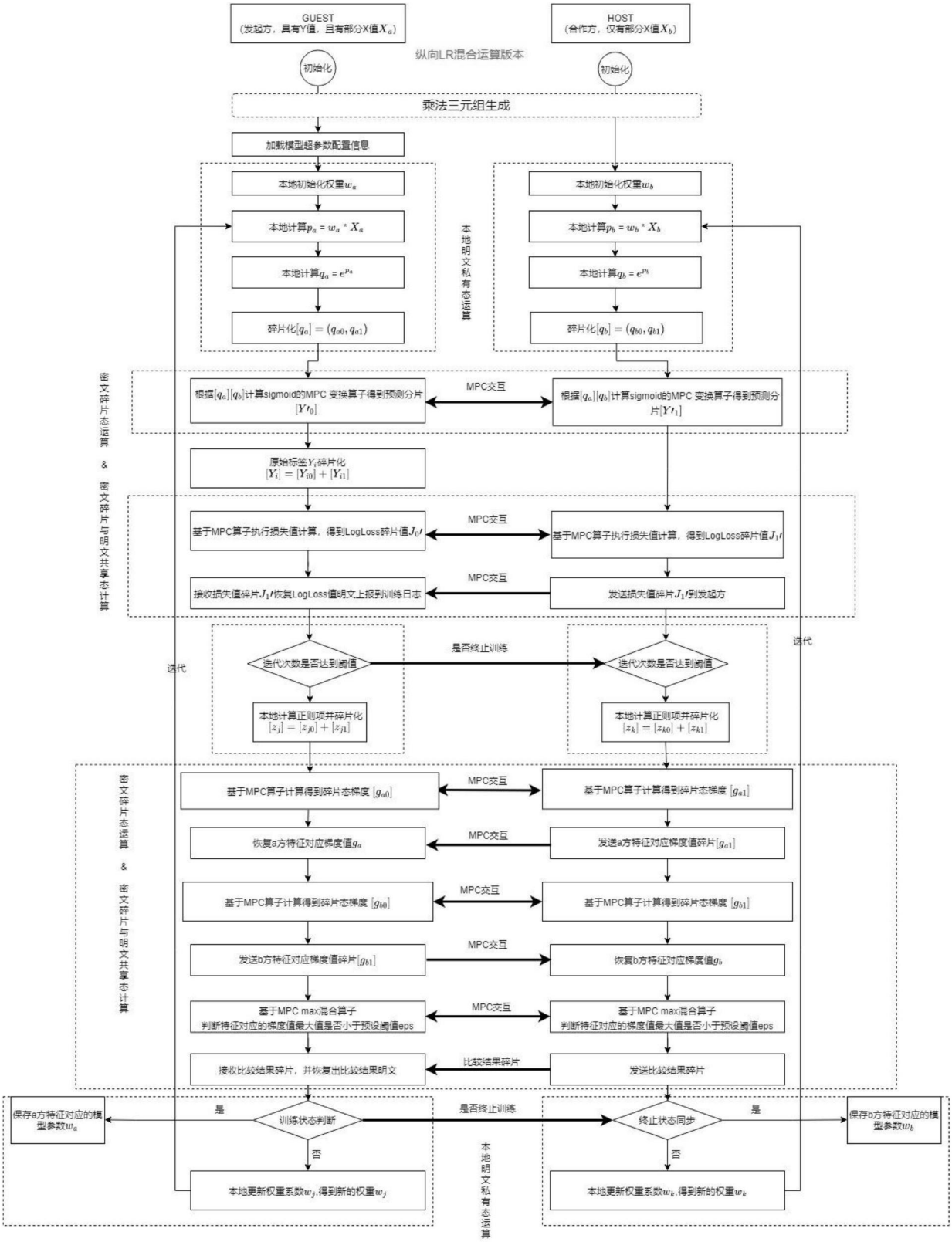

GUEST
（发起方，具有Y值，且有部分X值X_a）
HOST
（合作方，仅有部分X值X_b）
纵向LR混合运算版本
初始化
初始化
乘法三元组生成
加载模型超参数配置信息
本地初始化权重w_a
本地计算$p_a = w_a * X_a$
本地计算$q_a = e^{p_a}$
碎片化$[q_a] = (q_{a0}, q_{a1})$
本地初始化权重w_b
本地计算$p_b = w_b * X_b$
本地计算$q_b = e^{p_b}$
碎片化$[q_b] = (q_{b0}, q_{b1})$
本地明文私有态运算
密文碎片态运算 & 密文碎片与明文共享态计算
根据$[q_a][q_b]$计算sigmoid的MPC 变换算子得到预测分片$[Y'_0]$
MPC交互
根据$[q_a][q_b]$计算sigmoid的MPC 变换算子得到预测分片$[Y'_1]$
原始标签Y_i碎片化
$[Y_i] = [Y_{i0}] + [Y_{i1}]$
基于MPC算子执行损失值计算，得到LogLoss碎片值J_0'
MPC交互
基于MPC算子执行损失值计算，得到LogLoss碎片值J_1'
接收损失值碎片J_1'恢复LogLoss值明文上报到训练日志
MPC交互
发送损失值碎片J_1'到发起方
迭代
迭代次数是否达到阈值
是否终止训练
迭代次数是否达到阈值
迭代
本地计算正则项并碎片化
$[z_j] = [z_{j0}] + [z_{j1}]$
本地计算正则项并碎片化
$[z_k] = [z_{k0}] + [z_{k1}]$
密文碎片态运算 & 密文碎片与明文共享态计算
基于MPC算子计算得到碎片态梯度$[g_{a0}]$
MPC交互
基于MPC算子计算得到碎片态梯度$[g_{a1}]$
恢复a方特征对应梯度值g_a
MPC交互
发送a方特征对应梯度值碎片$[g_{a1}]$
基于MPC算子计算得到碎片态梯度$[g_{b0}]$
MPC交互
基于MPC算子计算得到碎片态梯度$[g_{b1}]$
发送b方特征对应梯度值碎片$[g_{b1}]$
MPC交互
恢复b方特征对应梯度值g_b
基于MPC max混合算子
判断特征对应的梯度值最大值是否小于预设阈值eps
MPC交互
基于MPC max混合算子
判断特征对应的梯度值最大值是否小于预设阈值eps
接收比较结果碎片，并恢复出比较结果明文
比较结果碎片
发送比较结果碎片
保存a方特征对应的模型参数w_a
是
训练状态判断
是否终止训练
终止状态同步
是
保存b方特征对应的模型参数w_b
否
否
本地更新权重系数w_j,得到新的权重w_j
本地更新权重系数w_k,得到新的权重w_k
本地明文私有态运算

图4

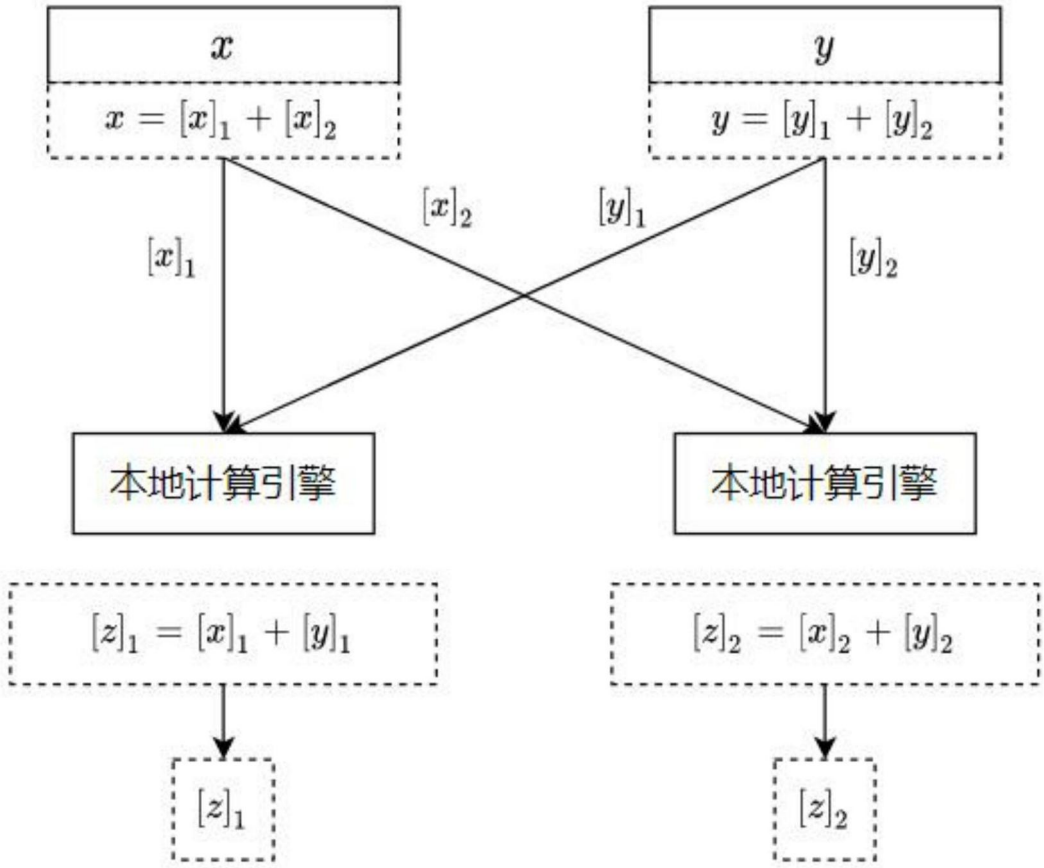


图5a

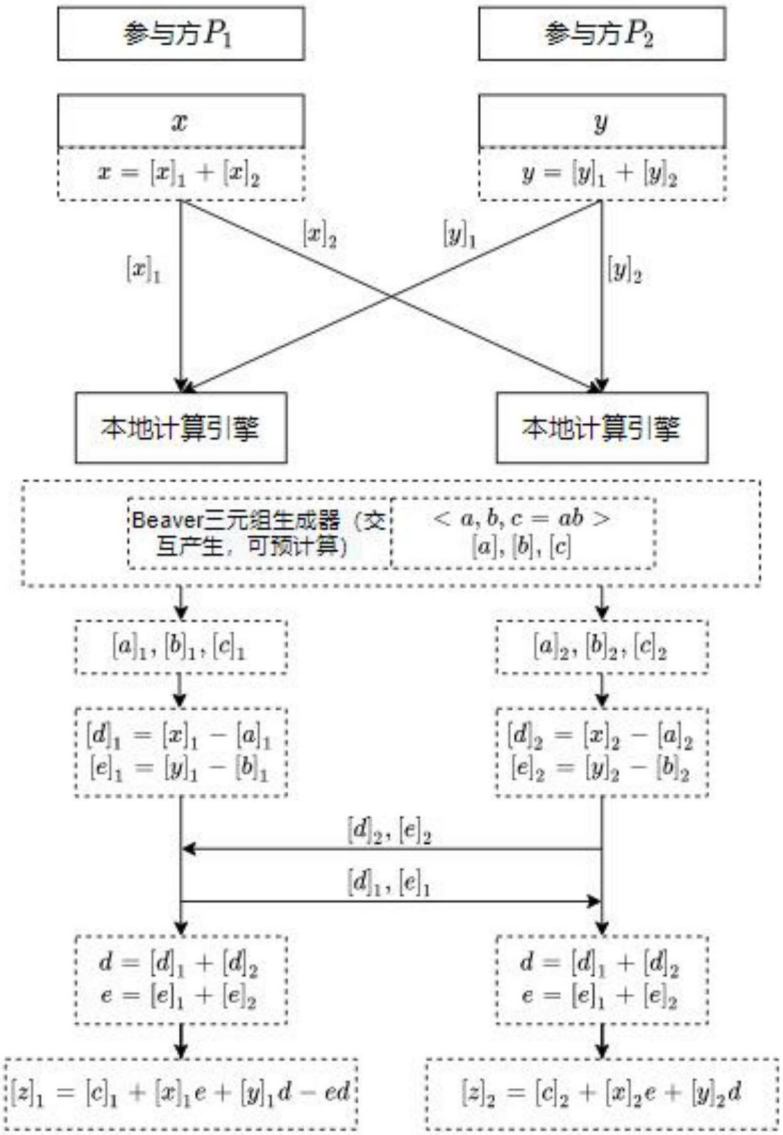

参与方P_1
参与方P_2
x
$x = [x]_1 + [x]_2$
y
$y = [y]_1 + [y]_2$
$[x]_1$
$[x]_2$
$[y]_1$
$[y]_2$
本地计算引擎
本地计算引擎
Beaver三元组生成器（交互产生，可预计算）
$< a, b, c = ab >$
$[a], [b], [c]$
$[a]_1, [b]_1, [c]_1$
$[a]_2, [b]_2, [c]_2$
$[d]_1 = [x]_1 - [a]_1$
$[e]_1 = [y]_1 - [b]_1$
$[d]_2 = [x]_2 - [a]_2$
$[e]_2 = [y]_2 - [b]_2$
$[d]_2, [e]_2$
$[d]_1, [e]_1$
$d = [d]_1 + [d]_2$
$e = [e]_1 + [e]_2$
$d = [d]_1 + [d]_2$
$e = [e]_1 + [e]_2$
$[z]_1 = [c]_1 + [x]_1 e + [y]_1 d - ed$
$[z]_2 = [c]_2 + [x]_2 e + [y]_2 d$

图5b

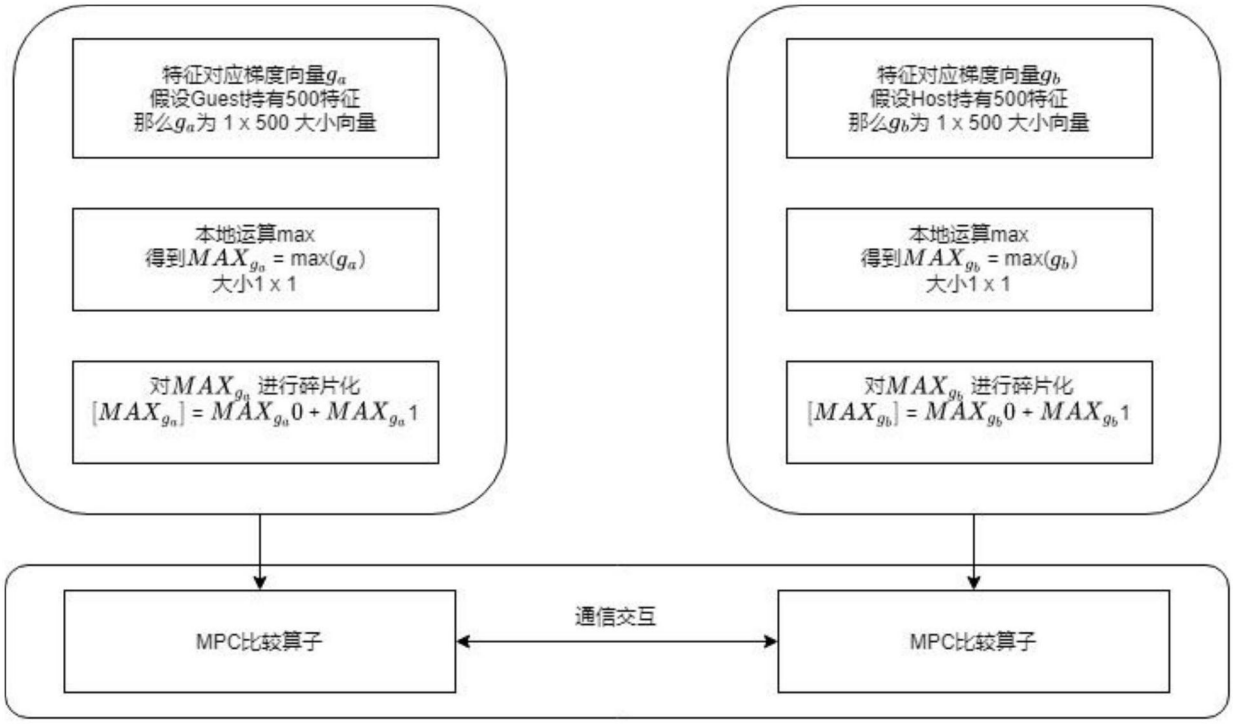

特征对应梯度向量g_a
假设Guest持有500特征
那么g_a为 1 x 500 大小向量
本地运算max
得到$MAX_{g_a} = \max(g_a)$
大小1 x 1
对MAX_{g_a} 进行碎片化
$[MAX_{g_a}] = MAX_{g_a}0 + MAX_{g_a}1$
特征对应梯度向量g_b
假设Host持有500特征
那么g_b为 1 x 500 大小向量
本地运算max
得到$MAX_{g_b} = \max(g_b)$
大小1 x 1
对MAX_{g_b} 进行碎片化
$[MAX_{g_b}] = MAX_{g_b}0 + MAX_{g_b}1$
MPC比较算子
通信交互
MPC比较算子

图5c

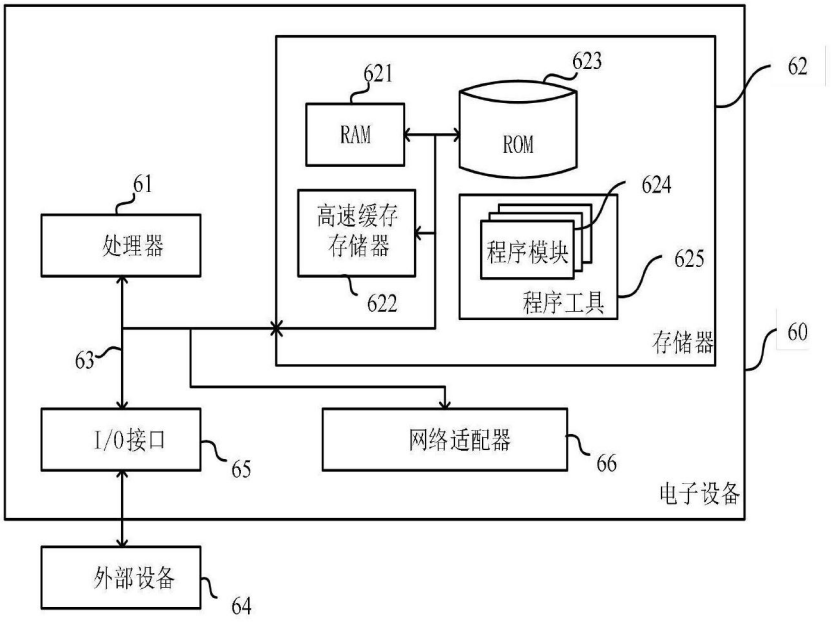

621
623
62
RAM
ROM
61
624
高速缓存
存储器
程序模块
625
622
程序工具
存储器
60
63
I/O接口
65
网络适配器
66
电子设备
外部设备
64

图6