US Patent & Trademark Office
Patent Public Search | Text View

United States Patent Application Publication	20250258959
Kind Code	A1
Publication Date	August 14, 2025
Inventor(s)	ZHANG; Yuanjian et al.

DATA PROCESSING METHOD AND APPARATUS, ELECTRONIC DEVICE, AND STORAGE MEDIUM

Abstract

The present application provides a data processing method and apparatus, an electronic device, and a storage medium, the method comprises: acquiring label residual values determined for respective data samples, each label residual value being used for representing a degree of deviation between a real label of a corresponding data sample and a label predicted by a target model; performing residual decomposition on the respective data samples based on respective label residual values to obtain model parameter correction amounts corresponding to the respective data samples; and sending the model parameter correction amounts to a data receiver, so that the data receiver reconstructs model parameter information of the target model based on the model parameter correction amounts. The present application can effectively perturb residuals through residual decomposition, which enables a data sender to protect the real labels owned thereby. At the same time, the data sender can send necessary correction information (i.e., model parameter correction amounts) to the data receiver to reconstruct accurate model parameter information, thus further ensuring the model training performance.

Inventors:	**ZHANG; Yuanjian (Shanghai, CN), ZHOU; Yongkai (Shanghai, CN), CHEN; Zhongzheng (Shanghai, CN)**
Applicant:	**CHINA UNIONPAY CO., LTD.** (Shanghai, CN)
Family ID:	**1000008618603**
Assignee:	**CHINA UNIONPAY CO., LTD. (Shanghai, CN)**
Appl. No.:	**19/104157**
Filed (or PCT Filed):	**January 10, 2023**
PCT No.:	**PCT/CN2023/071564**

Foreign Application Priority Data

CN	202210975564.1	Aug. 15, 2022

Publication Classification

Int. Cl.: **G06F21/62** (20130101)

U.S. Cl.:

CPC **G06F21/6245** (20130101);

Background/Summary

[0001] The present application claims priority for a Chinese patent application filed on Aug. 15, 2022, with application number of CN202210975564.1 and title of “Data Processing Method and Apparatus, Electronic Device, and Storage Medium”. The disclosure of the Chinese patent application is hereby incorporated by reference for all purposes.

TECHNICAL FIELD

[0002] The present application relates to the technical field of data security, and specifically to a data processing method and apparatus, an electronic device, and a storage medium.

BACKGROUND

[0003] With the development of application fields such as artificial intelligence and big data mining and analysis, the demand for data volume is ever increasing. To achieve mining of greater value, it is often necessary to integrate data from multiple parties. The multi-party data here may come from different organizations, for example, transaction data may be scattered across financial institutions, and medical diagnosis records may come from medical institutions. In addition, the multi-party data may also come from different industries, for example, transaction data may either come from e-commerce or finance.

[0004] Multi-party data elements are subject to regulatory compliance constraints during the process of circulation, making the approach of centrally collecting detailed data for model training for business operations unfeasible, which constitutes a data barrier. To break down the data barrier, the approach of Federated learning may be used to enable all parties involved to obtain a global model without the need to share their private data, where the private data here may be label information about the target user, such as whether the target user is a high net worth user. While reasonably mining the application value of data, it is also necessary to prevent data abuse and protect privacy data. In practical applications, model related information may be exchanged through encryption to achieve collaborative optimization of the Federated model.

[0005] A residual encryption method is provided in related technologies. Taking label data as privacy data as an example, the labeled party may make a differential privacy in the form of addition before sending the model related information, so that after the differential privacy conditions are reached, the labeled party may send it to the unlabeled party in the form of homomorphic encryption. Due to the presence of noise, the unlabeled party is unable to reconstruct the true residuals based on the model related parameters grasped, thus achieving the goal of privacy data protection.

[0006] However, due to the inherent noise in differential privacy, there is certain performance loss in model accuracy for the unlabeled party in the future.

SUMMARY

[0007] The embodiments of the present application provide at least a data processing method and apparatus, an electronic device, and a storage medium. The data sender encrypts the label, which is private data, by means of residual decomposition, and the data receiver reconstructs the model parameters based on the model parameter correction amounts obtained through encryption, resulting in a reconstructed model with relatively high accuracy.

[0008] According to a first aspect, an embodiment of the present application provides a data processing method, comprising: [0009] acquiring label residual values determined for respective data samples, each label residual value being used for representing a degree of deviation between a real label of a corresponding data sample and a label predicted by a target model; [0010] performing residual decomposition on the respective data samples based on respective label residual values to obtain model parameter correction amounts corresponding to the respective data samples; and [0011] sending the model parameter correction amounts to a data receiver, so that the data receiver reconstructs model parameter information of the target model based on the model parameter correction amounts.

[0012] In a possible embodiment, performing residual decomposition on the respective data samples based on the respective label residual values to obtain model parameter correction amounts corresponding to the respective data samples, comprises: [0013] ordering the respective label residual values in an order of data size to obtain respective ordered label residual values; [0014] changing the ordered label residual values where the target data sample is in on the condition that a target data sample that requires residual decomposition is selected from the respective data samples, to obtain residual change information corresponding to the respective data samples; and determining model parameter correction amounts corresponding to the respective data samples based on the residual change information corresponding to the respective data samples and the real labels of the respective data samples.

[0015] In a possible embodiment, the target data sample that requires residual decomposition is selected from the respective data samples according to the following steps: [0016] grouping the respective ordered label residual values according to a preset number of groups to obtain grouped label residual values; [0017] selecting a preset number of target label residual values from each grouped label residual values; and [0018] determining a data sample corresponding to the target label residual values as the target data sample.

[0019] In a possible embodiment, changing the ordered label residual values where the target data sample is in on the condition that the residual change information corresponding to the respective data samples corresponds to a residual change vector, to obtain the residual change information corresponding to the respective data samples, comprises: [0020] for each of the data samples, on the condition that it is judged that the data sample is not the target data sample, determining that the data sample corresponds to a first residual change value; or, [0021] on the condition that it is judged that the data sample is the target data sample and the label residual value of the data sample is greater than zero, determining that the data sample corresponds to a second residual change value; or, [0022] on the condition that it is judged that the data sample is the target data sample and the label residual value of the data sample is less than zero, determining that the data sample corresponds to a third residual change value; and [0023] collecting residual change values respectively corresponding to the respective data samples to determine the residual change vector; wherein, the residual change vector is used to represent whether the label residual values of the respective data samples have changed.

[0024] In a possible embodiment, determining model parameter correction amounts corresponding to the respective data samples based on the residual change information corresponding to the respective data samples and the real labels of the respective data samples, comprises: [0025] performing point multiplication operation on the residual change vector and a transpose result of the residual change vector to determine a first operator; and, determining a second operator corresponding to the label values on the condition that the label values corresponding to the real labels of the respective data samples are determined; and [0026] determining the model parameter correction amounts corresponding to the respective data samples based on the first operator and the second operator.

[0027] In a possible embodiment, after obtaining the residual change information corresponding to the respective data samples, the method further comprises: [0028] determining changed label residual values based on the residual change information corresponding to the respective data samples and product operation between the respective ordered label residual values; [0029] determining the model parameter information to be sent to the data receiver based on the changed label residual values; and [0030] sending the model parameter information to the data receiver.

[0031] In a possible embodiment, sending the model parameter information to the data receiver comprises: [0032] sending the model parameter information to the data receiver on the condition that model convergence conditions are not reached, [0033] sending the model parameter correction amounts to the data receiver comprises: [0034] sending the model parameter correction amounts to the data receiver on the condition that the model convergence conditions are reached.

[0035] According to a second aspect, the present application also provides a data processing method, comprising: [0036] receiving model parameter correction amounts corresponding to respective data samples sent by a data sender; wherein, the model parameter correction amounts are determined by means of residual decomposition based on label residual values determined for the respective data samples, each label residual value being used for representing a degree of deviation between a real label of a corresponding data sample and a label predicted by a target model; and [0037] reconstructing model parameter information of the target model based on the model parameter correction amounts.

[0038] In a possible embodiment, reconstructing the model parameter information of the target model based on the model parameter correction amounts comprises: [0039] acquiring data characteristic information input for the target model; and [0040] determining the model parameter information of the target model based on the data characteristic information and the model parameter correction amounts.

[0041] In a possible embodiment, on the condition that the data characteristic information includes a data characteristic vector, determining the model parameter information of the target model based on the data characteristic information and the model parameter correction amounts comprises: [0042] performing point multiplication operation on the data characteristic vector and a transpose result of the data characteristic vector to determine a third operator; and, performing point multiplication operation on the transpose result of the data characteristic vector and the data characteristic vector to determine a fourth operator; and [0043] determining the model parameter information of the target model based on the third operator, the fourth operator, and the model parameter correction amounts.

[0044] According to a third aspect, the present application also provides a data processing apparatus, comprising: [0045] an acquisition module, for acquiring label residual values determined for respective data samples, each label residual value being used for representing a degree of deviation between a real label of a corresponding data sample and a label predicted by a target model; [0046] a decomposition module, for performing residual decomposition on the respective data samples based on the respective label residual values to obtain model parameter correction amounts corresponding to the respective data samples; and [0047] a sending module, for sending the model parameter correction amounts to a data receiver, so that the data receiver reconstructs model parameter information of the target model based on the model parameter correction amounts.

[0048] According to a fourth aspect, the present application also provides a data processing apparatus, comprising: [0049] a receiving module, for receiving model parameter correction amounts corresponding to respective data samples sent by a data sender; wherein, the model parameter correction amounts are determined by means of residual decomposition based on label residual values determined for the respective data samples, each label residual value being used for representing a degree of deviation between a real label of a corresponding data sample and a label

predicted by a target model; and [0050] a reconstruction module, for reconstructing model parameter information of the target model based on the model parameter correction amounts.
[0051] According to a fifth aspect, the present application also provides an electronic device, comprising: a processor, a memory, and a bus, the memory stores machine-readable instructions executable by the processor, when the electronic device is running, the processor and the memory communicate with each other through the bus, and when the machine-readable instructions are executed by the processor, the data processing method according to any of the first aspect and its various embodiments, and the second aspect and its various embodiments is executed.
[0052] According to a sixth aspect, the present application also provides a computer-readable storage medium with a computer program stored thereon, when the computer program is run by a processor, the data processing method according to any of the first aspect and its various embodiments, and the second aspect and its various embodiments is executed.
[0053] By using the above data processing method and apparatus, electronic device, and storage medium, on the condition that label residual values determined for the respective data samples are obtained, residual decomposition can be performed on the respective data samples based on the respective label residual values. Then, the model parameter correction amounts corresponding to the respective data samples obtained by means of residual decomposition can be sent to a data receiver, and the data receiver can reconstruct the model parameter information based on the model parameter correction amounts. In the present application, the data sender (i.e., the labeled party) can effectively perturb the residuals during the residual decomposition process, so that the data receiver (i.e., the unlabeled party) can only construct a model with erroneous label information. This enables the data sender to protect the real labels owned thereby, and at the same time, the labeled party can transmit necessary correction information (i.e., model parameter correction amounts) to the unlabeled party, allowing the unlabeled party to reconstruct accurate model parameter information based on the correction information so as to ensure the model training performance.
[0054] Other advantages of the present application will be described in further detail in conjunction with the following description and accompanying drawings.
[0055] It should be appreciated that the above description is only a summary of the technical solution of the present application in order to provide a clearer understanding of the technical means of the present application, so that the technical solution can be implemented according to the content of the specification. In order to make the above and other purposes, features, and advantages of the present application more clear and understandable, the following examples are specifically provided to illustrate the specific embodiments of the present application.

Description

BRIEF DESCRIPTION OF THE DRAWINGS
[0056] In order to describe the technical solution of the embodiments of the present application more clearly, the accompanying drawings required for use in the embodiments will be introduced briefly. The accompanying drawings are incorporated into the specification and form a part of it. These accompanying drawings illustrate embodiments that comply with the present application and are used together with the specification to explain the technical solution of the present application. It should be appreciated that the following accompanying drawings only illustrate certain embodiments of the present application and therefore should not be considered as limiting the scope. For those skilled in the art, other relevant accompanying drawings may be obtained based on these accompanying drawings without creative labor. In addition, throughout the accompanying drawings, the same reference numerals are used to denote the same components. In the accompanying drawings:

[0057] FIG. **1** shows a flowchart of a data processing method provided by an embodiment of the present application;

[0058] FIG. **2** shows a flowchart of a specific method for determining residual change information in a data processing method provided by an embodiment of the present application;

[0059] FIG. **3** shows a timing flowchart of a data processing method provided by an embodiment of the present application;

[0060] FIG. **4** shows a schematic diagram of the application of a data processing method provided by an embodiment of the present application;

[0061] FIG. **5** shows a flowchart of another data processing method provided by an embodiment of the present application;

[0062] FIG. **6** shows a schematic diagram of a data processing apparatus provided by an embodiment of the present application;

[0063] FIG. **7** shows a schematic diagram of another data processing apparatus provided by an embodiment of the present application; and

[0064] FIG. **8** shows a schematic diagram of an electronic device provided by an embodiment of the present application.

DETAILED DESCRIPTION

[0065] The exemplary embodiments of the present application will be described in further detail below with reference to the accompanying drawings. Although the exemplary embodiments of the present application are shown in the accompanying drawings, it should be appreciated, however, that the present application may be implemented in various forms and should not be limited by the embodiments described herein. On the contrary, these embodiments are provided here to enable a more thorough understanding of the present application and to fully convey the scope of the present application to those skilled in the art.

[0066] In the description of the embodiments of the present application, it should be appreciated that terms such as “include” or “have” are intended to indicate the existence of the features, numbers, steps, actions, components, parts, or combinations thereof disclosed in this specification, and are not intended to exclude the possibility of the existence of one or more other features, numbers, steps, actions, components, parts, or combinations thereof.

[0067] Unless otherwise specified, “/” means “or”, for example, A/B may represent A or B. “And/or” herein is only a description of the association relationship between associated objects, indicating that there may be three types of relationships, for example, A and/or B may represent: A exists alone, A and B exist simultaneously, and B exists alone.

[0068] The terms “first”, “second”, etc. are only used for descriptive purposes, and cannot be understood as indicating or implying relative importance or implying or indicating the number of technical features indicated. Therefore, features limited by “first” and “second” may explicitly or implicitly include one or more of the features. In the description of the embodiments of the present application, unless otherwise specified, “a plurality of” means two or more.

[0069] It has been found upon research that in related technologies, model related information may be exchanged by means of encryption to achieve collaborative optimization of a Federated model.

[0070] A residual encryption method is provided in related technologies. Taking labeled data as privacy data as an example, the labeled party may make a differential privacy in the form of addition before sending model related information, so that after the differential privacy conditions are reached, the labeled party may send it to the unlabeled party in the form of homomorphic encryption.

[0071] Wherein, in the process of performing differential privacy, that is, adding noise to the residual r.sub.i to make the differential privacy to be: {tilde over (r)}.sub.i=r.sub.i+noise, wherein the added noise satisfies the condition noise˜Lap(2ε.sup.−1) (i.e., the noise comes from the Laplace random variable generated by the standard deviation of 2ε.sup.−1). Due to the presence of noise, the unlabeled party is unable to reconstruct the real residual r.sub.i based on the model related parameters grasped, thus achieving the goal of protecting privacy data.

[0072] Another encryption method is localized differential privacy random response. The idea is to randomly perturb the composition of a batch, so that the dimensions of interaction residuals remain unchanged in terms of form, but the actual rank is less than the number of samples in the batch, resulting in residuals being not unique.

[0073] Specifically, the labeled party selects a subset C from the samples of the labeled party based on the batch size agreed by both parties, generates a random response vector m about the label y={0,1} using category distribution of the labels, and sends m.sub.rr obtained by differential privacy processing to the unlabeled party. When the unlabeled party constructs a local model based on the feature x.sup.b, the labeled party gives the real residual value r.sub.i of the residual m.sub.i=1 obtained from the random response, while setting 0 for samples that are not in the batch. While ensuring that the residual dimension of the interaction is still the size of the batch (|B|), the constructed residual matrix is sent to the unlabeled party after homomorphic encryption. The formal representation of the composition of the sent content D is:

[00001] $$\text{.Math. } r_i \text{ .Math. } = \{ \begin{matrix} \text{.Math.} & (l_i)\text{ .Math.} & i \in D \\ & 0 & i\text{ .Math. } D \end{matrix}$$

wherein, D⊂{i:m.sub.i=1}{circumflex over ( )}|D|=|B|, σ(•) is the sigmoid function, and custom-character•custom-character is homomorphic encryption. This method is called a label protection strategy mixing differential privacy with homomorphic encryption, and can achieve & differential privacy under the following conditions:

[00002] $$\frac{q\exp(\ )}{1+\exp(\ )} + \frac{1-q}{1+\exp(\ )} < 1, 0 < q < \frac{1}{2}$$

q is the proportion marked as 1 in m. By the above operations by the labeled party, the unlabeled party is unable to construct a linear equation set about the residuals.

[0074] However, although the above method achieves label protection and reduces the accuracy loss of classification compared to the first encryption method, it does not fully utilize the sample information in the batch due to the existence of differential privacy and random zero setting. Therefore, there is still a performance loss compared to plaintext computation, and the reduction of accuracy loss of classification is at the cost of increasing training costs, which is not likely to be put into production use in the short term.

[0075] In order to at least partially address one or more of the aforementioned issues and other potential problems, the present application provides at least a data processing solution. The labeled party encrypts the label, which is private data, by means of residual decomposition, while the unlabeled party performs model parameter reconstruction based on the model parameter correction amounts obtained through encryption, resulting in a reconstructed model with relatively high accuracy.

[0076] It should be noted that both labeled and unlabeled parties in the embodiments of the present application require local computation, and correct information in a collaborative and interactive manner, with no significant increase in computational costs. The labeled party here corresponds to the data sender, the unlabeled party corresponds to the data receiver, residual decomposition is performed on the side of the labeled party, and parameter reconstruction is completed on the side of the unlabeled party.

[0077] In practical applications, the data sender here refers to the initiator with real labels and its processing equipment used to perform operations including residual decomposition, and the data receiver here refers to the responder with model input features but without real labels and its processing equipment used to perform parameter reconstruction operations.

[0078] In order to facilitate the understanding of the embodiments, a data processing method disclosed in embodiments of the present application will first be described in detail. The executing subject of the data processing method provided in embodiments of the present application is generally an electronic device with certain computing power. The electronic device includes, for example, terminal devices or other processing devices. The terminal devices may be user equipment (UE), mobile devices, user terminals, cellular phones, personal digital assistants (PDA), handheld devices, and the like. In some possible embodiments, the data processing method may be implemented by the processor calling computer-readable instructions stored in the memory.

[0079] As shown in FIG. **1**, it is a flowchart of a data processing method provided by an embodiment of the present application. The method is mainly executed by the data sender, and specifically includes steps S**101**-S**103**, wherein: [0080] S**101**: acquiring label residual values determined for respective data samples, each label residual value being used for representing a degree of deviation between a real label of a corresponding data sample and a label predicted by a target model; [0081] S**102**: performing residual decomposition on the respective data samples based on respective label residual values to obtain model parameter correction amounts corresponding to the respective data samples; and [0082] S**103**: sending the model parameter correction amounts to a data receiver, so that the data receiver reconstructs model parameter information of the target model based on the model parameter correction amounts.

[0083] In order to facilitate the understanding of the data processing method provided by the embodiments of the present application, the application scenarios of the method will first be described in detail. The data processing method in the embodiments of the present application may mainly be applied to the field of Federated learning, where Federated learning here may correspond to a fusion learning approach of data from multiple parties, for example, it may be the fusion data formed from transaction data from financial institutions and medical diagnosis records from medical institutions.

[0084] In related technologies, the vertical logistic regression method is mostly used to construct joint models. However, due to the inherent characteristic constraints of the vertical logistic regression method, the private data of one party is likely to be inferred by the other party. For example, the label data generated within financial institutions regarding whether a user is a high net worth user is likely to leak to medical institutions with the sharing of model parameter information, which will bring certain insecurity factors to user privacy.

[0085] In order to enhance the protection of privacy data while better promoting Federated learning, the embodiments of the present application provide a collaborative data processing scheme, which mainly corresponds to two phases, namely, model training phase and model correction phase.

[0086] For the model training phase, it is mainly the process of performing residual decomposition by the labeled party. After decomposition, only a portion of the residual related confusion information is exchanged with the unlabeled party under homomorphic encryption protection, while another portion of the residual retention information is held by the labeled party and does not participate in the model training process. As such, even if the unlabeled party solves the residuals according to the relevant label leakage attack approach (satisfying the column full rank condition), as the residuals no longer have a one-to-one correspondence with the real labels, the accuracy in inferring the labels of the labeled party by the unlabeled party is almost the same as that of random guessing binary classification results. Therefore, it is impossible to infer the real labels of the labeled party based on the sign of the residual obtained by known label leakage attack method, thus ensuring the security of privacy data.

[0087] For the model correction phase, it is mainly the process of reconstructing parameters by the unlabeled party. After the convergence of model training by the labeled party, initiated by the labeled party and responded by the unlabeled party, only an additional round of interaction between the labeled party and the unlabeled party about the weight information of the unlabeled party is needed to complete the correction of each characteristic weight of the unlabeled party. Due to the fact that the additional information provided by the labeled party cannot be obtained during the model training phase, and this information is the entirety of residual retention implemented by the real label and the protected label, the unlabeled party cannot obtain the label information of the labeled party during the model correction phase even though the unlabeled party reconstructs the model with high accuracy, thereby further ensuring the security of private data.

[0088] In an embodiment of the present application, the data sender, upon acquiring the label residual values determined for the respective data samples, may perform residual decomposition on the respective data samples based on the respective label residual values to obtain the model parameter correction amounts corresponding to the respective data samples, and finally send the model parameter correction amounts to the data receiver. The model parameter correction amounts here refer to the additional information provided by the labeled party to the unlabeled party, where the additional information is determined after decomposing the residuals.

[0089] The labeled party randomly performs residual decomposition locally based on the fitting degree of the unlabeled party to its own labels, in order to achieve the effect of randomly replacing label information, so that the unlabeled party cannot infer the real labels based on the residual information processed by the labeled party.

[0090] It can be seen that in the embodiments of the present application, by perturbing the residuals, the unlabeled party can only construct the model with erroneous label information, and the contained erroneous information is generated by the labeled party's local decomposition of the residual information, which the unlabeled party cannot distinguish. Then, the labeled party transmits necessary correction information to the unlabeled party, so that the unlabeled party can correct the weights of the characteristics, but cannot obtain the labels of the labeled party.

[0091] The unlabeled party corrects the biased characteristic weights with the cooperation of the labeled party to obtain the correct model parameter information. Due to the fact that this information includes the labels of the labeled party and the perturbation information on the labels, where the perturbation information is independent of the information known to the unlabeled party during the model training phase, the unlabeled party cannot obtain the label information of the labeled party.

[0092] Considering the crucial role of determining the model parameter correction amounts in the reconstruction of model parameters by the unlabeled party, the process of determining the model parameter correction amounts may be emphasized, which may be achieved through the following steps: [0093] Step 1: ordering the respective label residual values in an order of data size to obtain respective ordered label residual values; [0094] Step 2: changing the ordered label residual values where the target data sample is in on the condition that a target data sample that requires residual decomposition is selected from the respective data samples, to obtain residual change information corresponding to the respective data samples; and [0095] Step 3: determining model parameter correction amounts corresponding to the respective data samples based on the residual change information corresponding to the respective data samples and the real labels of the respective data samples.

[0096] Here, the respective label residual values may first be ordered and the residual change information may be obtained, and then the model parameter correction amounts may be determined based on the residual change information and the real labels.

[0097] Wherein, the above residual change information is used to indicate whether the label residual values have changed. The changed label residual values may be marked with a first label, and the unchanged label residual values may be marked with a second label, so as to achieve the purpose of randomly replacing the real labels. Also considering that the model parameter correction amounts are obtained by combining the residual change information and the real labels, that is, what the unlabeled party obtains are not the direct real labels but the correction amounts after label mapping. Based on the correction amounts and the corresponding input features, the model parameter information may be constructed.

[0098] It should be noted that the labeled party only needs to map the real labels and the made labels locally for computation and synthesis after the model converges, and then send to the unlabeled party.

[0099] In order to achieve the purpose of randomly replacing real labels, it is necessary to select the target sample through a series of operations such as grouping and selection before performing residual changes. Specifically, the respective ordered label residual values may first be grouped according to a preset number of groups to obtain the grouped label residual values. Then, a preset number of target label residual values may be selected from each grouped label residual values, and finally, the data sample corresponding to the target label residual values is determined as the target data sample. The grouping here may be preset, for example, may be divided into two groups, four groups, etc., where no specific restrictions is made in this regard, with the purpose of preventing the unlabeled party from obtaining additional information about the label information of the labeled party based on the similarity between characteristics. At the same time, in the process of selecting data samples that require residual change from the respective groups, random quantitative extraction may be used to ensure the randomness of label replacement, thus further enhancing data security.

[0100] The residual change information in embodiments of the present application may refer to the residual change vector, each vector corresponding to the residual change situation of a data sample. Wherein, on the condition that it is judged that a data sample is not the target data sample, it may be determined that the data sample corresponds to a first residual change value; on the condition that it is judged that a data sample is the target data sample and the label residual value of the data sample is greater than zero, it may be determined that the data sample corresponds to a second residual change value; on the condition that it is judged that a data sample is the target data sample and the label residual value of the data sample is less than zero, it may be determined that the data sample corresponds to a third residual change value. Then, the residual change values corresponding to the respective data samples may be collected to determine the residual change vector.

[0101] Based on the residual change vector mentioned above, the model parameter correction amounts may be determined by the following steps: [0102] Step 1: performing point multiplication operation on the residual change vector and a transpose result of the residual change vector to determine a first operator; and, determining a second operator corresponding to the label values on the condition that the label values corresponding to the real labels of the respective data samples are determined; and [0103] Step 2: determining the model parameter correction amounts corresponding to the respective data samples based on the first operator and the second operator.

[0104] Here, the model parameter correction amounts may be determined based on the first operator and the second operator. The model parameter correction amounts may also be a multi-dimensional vector, where each dimension points to the correction amount of a data sample, thereby achieving overall correction for the target model.

[0105] In the process of determining the residual change information, the product operation between the residual change information corresponding to the respective data samples and the respective ordered label residual values may also be used to determine the changed label residual values. Then, the model parameter information to be sent to the data receiver may be determined based on the changed label residual values, and the model parameter information may be sent to the data receiver.

[0106] Wherein, the computation process regarding the changed label residual values mentioned above may be a step that only needs to be executed once before the model converges. For example, the labeled party may perform the above step in the first round to change the residual values of the randomly selected samples, the effect of which is equivalent to changing the real label of the sample. On the premise of ensuring label security, it also reduces the computational cost.

[0107] It should be noted that the model parameter information here may be sent from the labeled party to the unlabeled party during the intermediate training process, that is, before reaching the model convergence conditions. Due to residual changes, even if the unlabeled party receives the model parameter information, it cannot infer the real label. After completing the model training, that is, when the model convergence conditions are reached, the labeled party may send the correction amounts to the unlabeled party to facilitate the reconstruction of accurate model parameter information by the unlabeled party.

[0108] In practical applications, model convergence may be associated with the training batch. When all data samples in a batch have completed one round of parameter updates, it may be determined whether the model reaches the convergence conditions. If it converges, the correction amounts are sent in the above way. Otherwise, the model training continues.

[0109] To facilitate the understanding of the process of determining the model parameter correction amounts mentioned above, further explanation may be provided in conjunction with the following embodiments.

[0110] Firstly, an example of the prerequisite is as follows: [0111] 1) Modeling needs to be completed between two parties, with Party A assuming the role of data application party (Guest), and Party B assuming the roles of data provider (Host) and collaborator (Arbiter). The label is on the side of Party A (i.e., labeled party), and the characteristic is on the side of Party B (i.e., unlabeled party); [0112] 2) The label of Party A is binary classification (0/1); [0113] 3) Party B only knows that Party A's label is in the form of binary classification 0/1, and other information about Party A's label is unknown to Party B, but the labels of either category are of interest to Party B (i.e., Party B wants to know the label information of any sample of Party A); [0114] 4) Both parties agree to complete the joint modeling according to the basic steps of vertical logistic regression, and Party B may only launch the semi-honest attacks on Party A.

[0115] Here, both labeled and unlabeled parties agree in a secure manner to use the vertical logistic regression for joint modeling, determining the size |B| and composition of the batch B of each round of modeling, model parameter learning rate η, and determining align samples through secure intersection. Wherein, Party B knows the characteristics {(x.sub.i)} of the align samples, and Party A knows the labels {y.sub.i} of the align samples. Each round is initiated by Party B, responded by Party A, and ends with Party B updating the parameters of the local model w.sub.b.sup.B(t). At the same time, Party A and Party B agree that after the model converges (flag=1), both parties collaboratively complete to reconstruct the true weight W of Party B.

[0116] During the model training phase, when the model has not reached the convergence conditions (at which point flag=0), the following steps are executed: [0117] In a batch B, Party B constructs a local model {l.sub.i.sup.B(t)} based on its own model parameters w.sub.b.sup.B(t) and characteristics set {(x.sub.i.sup.B(t))} (calculated using l.sub.i.sup.B(t)=x.sub.i.sup.Bw.sub.b.sup.B(t)), and then sends {l.sub.i.sup.B(t)} to Party A using homomorphic encryption (such as paillier algorithm).

[0118] When the model reaches the convergence conditions (at which point flag=0), both parties have completed the training of the model with erroneous labels and perform the following operations: [0119] Party A sends the model parameter correction amounts [00003]$\frac{y+1}{2}SS^T(SS^T)^{-1}$ to Party B, and Party B reconstructs the real model parameter w as follows:

[00004]$w=\frac{x^T(xx^T)^{-1}+(x^Tx)^{-1}x^T(\frac{y+1}{2}ss^T(ss^T)^{-1})}{2}$

wherein, X corresponds to the data characteristic vector, y corresponds to the value of the real label, S is the label mapping vector obtained by batch splicing of S.sub.B and corresponds to the residual change information.

[0120] It should be noted that the residual change information mentioned above may correspond to the retention information ¬r.sub.i.sup.B of the residuals in this batch of samples, and this retention information may be determined in the first round of training. It may be specifically implemented according to the following steps: [0121] 1) Party A orders all samples within the batch based on residuals r.sub.i.sup.B(t) (from small to large or from large to small). [0122] 2) Party A performs randomized residual decomposition on the residuals of this batch of samples as follows, and randomly select a number m∈{2, 4} as the group size composed of samples within this batch.

[0123] For the ordered sample sequence, starting from the first sample, every m samples form a group G.sub.i, i=1, 2, . . . , ┌|B|/m┐, where the symbol ┌|B|/m┐ represents the smallest integer not less than |B|/m. For any formed group G.sub.j, ┌|G.sub.j|/2┐ samples are randomly selected from the group to form a candidate set C.sup.B, and C.sup.B≜f(B,m) And, for all samples in, C.sup.B, the corresponding retention information ¬r.sub.i.sup.B is determined in the following manner.

[0124] Wherein, if r.sub.i.sup.(t)>0, then ¬r.sub.i.sup.B=1; If r.sub.i.sup.B(t)<0, then ¬r.sub.i.sup.b=−1, regardless of whether it is 1 or −1, residual change will be performed subsequently. For all samples x; that are not in C.sup.B, ¬r.sub.i.sup.B=0, residual change will not be performed subsequently. It may be specifically determined according to the following formula:

[00005]$$\neg r_i^B = \{ \begin{matrix} 1 & \text{sort}(i) \in C^{B(t)} \text{ .Math. } r_i^{B(t)} > 0 \\ 0 & \text{sort}(i) \text{ .Math. } C^{B(t)} \\ -1 & \text{sort}(i) \in C^{B(t)} \text{ .Math. } r_i^{B(t)} < 0 \end{matrix}$$

wherein, sort(i) is the order of the i-th sample obtained by residual Δr.sub.i.sup.B ordering in the batch. Here, in order to prevent Party B from inferring, it is possible to reorder according to the original order for the data samples by Party B.

[0125] In order to facilitate further illustration regarding the residual decomposition corresponding to the reorder operation, the residual decomposition diagram shown in FIG. **2** may be used in conjunction for further explanation.

[0126] As shown in FIG. **2**, with batch size |B|=8, after ordering the residual values in ascending order, the bold letters in the second and fourth columns are respectively non-zero representations of samples randomly selected by group by ¬r.sub.i.sup.B(t), when m=2 and m=4. The corresponding randomized residual decomposition may be seen in the 1/−1 parts of the third and fifth columns.

[0127] Here, residual change vector S.sup.B may be determined based on label retention information ¬r.sub.i.sup.B(t). Each element in S.sup.B reflects whether the residual information of the sample changes during the first round of residual decomposition. When the residual changes, S.sup.B=−1; otherwise S.sub.i.sup.B=1. Namely:

[00006]$$S_i^B = \{ \begin{matrix} -1 & \neg r_i^B \neq 0 \\ 1 & \neg r_i^B = 0 \end{matrix}$$

On the condition that the residual change vector S.sup.B is determined, point multiplication operation may be performed on the residual change vector and a transposition result of the residual change vector to determine a first operator SS.sup.T; and, on the condition that the label values corresponding to the real labels of the respective data samples are determined, a second operator

[00007]$\frac{y+1}{2}$

corresponding to the label value is determined, and then the model parameter correction amounts

[00008]$(\frac{y+1}{2} SS^T (SS^T)^{-1})$

is determined.

[0128] It should be noted that the above model parameter correction amounts may be sent by Party A to Party B after the model training is completed. During the model training process, the pseudo residual sent by Party A to Party B may be computed according to the following formula:

[00009]$\tilde{r}_i^{B(t)} = r_i^{B(t)} - \neg r_i^B$

In this way, Party B updates its own model parameter w.sub.b.sup.B(t) in the cryptographic state, and the computation method is:

[00010]$w_b^{B(t)} = w_b^{B(t)} - \tilde{g}_b^{B(t)}$

wherein, g.sub.b.sup.B(t) is the gradient information about the parameter w.sub.b.sup.B(t) obtained by Party B based on identity information, this information is equivalent to the pseudo gradient information {x.sub.i{tilde over (r)}.sub.i.sup.B(t)} obtained by Party A by multiplying the characteristics and corresponding pseudo residuals on each sample.

[0129] The above steps are repeated for each batch B until all samples have completed a round of parameter updates and it is then determined whether the model reaches the convergence conditions. If it reaches the convergence conditions, set flag=1, otherwise continue. The timing diagram of the main operations is shown in FIG. **3**.

[0130] It is known that an embodiment of the present application performs two steps of one residual decomposition (only in the first round of joint modeling) and one model parameter reconstruction information transmission on the side of the labeled party, while one weight construction is required on the side of the unlabeled party, which may be illustrated with reference to FIG. **4**.

[0131] As shown in FIG. **4**, after completing batch initialization, the unlabeled party may build a local model based on the model parameters provided by the labeled party, and then the labeled party determines whether it is in the first round of training. If so, the labeled party may construct retention residual and label transformation according to the above steps, while the unlabeled party may update based on pseudo residuals. Then, when it is determined that the model has converged, the labeled party may send additional information to the unlabeled party to facilitate weight reconstruction by the unlabeled party.

[0132] It should be noted that the embodiments of the present application do not limit the specific implementation methods of the two modules of batch initialization and reaching convergence conditions, and do not limit the implementation methods of the alignment of samples between both parties before batch initialization. The embodiments of the present application focus on the implementation of residual decomposition for the unlabeled party by the labeled party and the collaborative implementation of weight reconstruction with unlabeled party. Other step computations in the model training phase may be solved using classical gradient descent method.

[0133] In addition, data security may be analyzed from the two parts of model training and model correction. During the model training process, residual retention information is solely owned by the labeled party and does not participate in the model training process. The unlabeled party cannot determine whether the labeled party has modified a specific example label based on known information.

[0134] For labeled party, this kind of security is at the level of information theory, with the effect of making it impossible for the unlabeled party to obtain its own labels by increasing computing power, and only one round of local operations is required to be executed. And, the additional computational cost is negligible compared to the overall joint modeling cost. Therefore, the unlabeled party cannot obtain the real label y of the labeled party based on semi-honest attacks.

[0135] For the weight reconstruction part, the theoretical basis is the equation XWS=YS. Since the real label and its transformation information

[00011]$((\frac{y+1}{2}SS^{T}(SS^{T})^{-1}))$

are sent to the unlabeled party as a whole, this process does not involve gradient information interaction and is independent of known label leakage methods. This makes it impossible for the unlabeled party to compute the label mapping S made by labeled party for the labels and the actual label Y. Therefore, if attackers continue to use known label leakage attack methods, the random correspondence between residuals and labels cannot be eliminated from beginning to end. In other words, regardless of whether the unlabeled party obtains positive or negative residual values, their corresponding real labels may be positive (1) or negative (0).

[0136] Furthermore, the usability of the model may be analyzed from the perspective of weight reconstruction. Given that the background of the problem may be limited to binary classification, if the labeled party changes the residual values of the randomly selected samples in the first round, the effect is equivalent to changing the real labels of the samples, from positive class (negative class) to negative class (positive class). The significance of ordering lies in combating the leakage of more label information of the labeled party by the unlabeled party due to differences in random initialization parameters. Therefore, there is no significant difference in information when applying S or {¬r.sub.i.sup.B(t)} onto the real labels Y, and the information in the final results obtained are not obfuscated, including differential privacy, before being sent by the labeled party. The weights can theoretically achieve better results in terms of their ability to differentiate between sample categories.

[0137] As shown in FIG. **5**, it is a flowchart of a data processing method provided according to an embodiment of the present application. This method is mainly executed by the data receiver, which specifically includes steps S**501**-S**502**, where: [0138] S**501**: Receiving model parameter correction amounts corresponding to respective data samples sent by a data sender; wherein, the model parameter correction amounts are determined by means of residual decomposition based on label residual values determined for the respective data samples, each label residual value being used for representing a degree of deviation between a real label of a corresponding data sample and a label predicted by a target model; [0139] S**502**: Reconstructing model parameter information of the target model based on the model parameter correction amounts.

[0140] Here, parameter reconstruction of the target model may be achieved based on the received model parameter correction amounts. Since the model parameter correction amounts here are determined based on the real labels of the data sender and the corresponding residual change information, the corresponding label replacement situation may be marked. And, the real labels cannot be resolved based on the correction amounts. Therefore, accurate reconstruction of model parameters may be achieved on the premise of protecting the real labels of the data sender.

[0141] Wherein, the process of determining the model parameter correction amounts may refer to the relevant description of the above embodiments, and will not be repeated here.

[0142] In the process of reconstructing model parameter information, in addition to relying on the above-mentioned model parameter correction amounts, it is also necessary to consider the data characteristic information input for the target model. This may specifically be achieved through the following steps: [0143] Step 1: Performing point multiplication operation on the data characteristic vector and a transpose result of the data characteristic vector to determine a third operator; and, performing point multiplication operation on the transpose result of the data characteristic vector and the data characteristic vector to determine a fourth operator; [0144] Step 2: Determining the model parameter information of the target model based on the third operator, the fourth operator, and the model parameter correction amounts.

[0145] The process of determining model parameter information in an embodiment of the present application may refer to the formula for model parameter w mentioned above:

[00012]$w = \frac{x^T(xx^T)^{-1} + (x^Tx)^{-1}x^T(\frac{y+1}{2}ss^T(ss^T)^{-1})}{2}$

XX.sup.T here corresponds to the third operator, X.sup.TX corresponds to the fourth operator,

[00013]$(\frac{y+1}{2}SS^T(SS^T)^{-1})$

corresponds to the correction amounts, and then the model parameters may be determined.

[0146] In the description of the present specification, the expressions such as referring to the terms “some possible embodiments”, “some embodiments”, “examples”, “specific examples”, or “some examples” mean that the specific features, structures, materials, or characteristics described in conjunction with the embodiments or examples are included in at least one embodiment or example of the present application. In the present specification, the schematic expressions of the above terms do not necessarily refer to the same embodiments or examples. And, the specific features, structures, materials, or characteristics described may be combined in any one or more embodiments or examples in an appropriate manner. In addition, those skilled in the art may integrate and combine the different embodiments or examples described in the present specification, as well as the features of different embodiments or examples, without conflicting with each other.

[0147] Regarding the flowcharts of the methods of embodiments of the present application, some operations are described as different steps executed in a certain order. Such flowcharts are illustrative rather than restrictive. Some steps described herein may be grouped together and executed in a single operation, some steps may be divided into a plurality of sub-steps, and some steps may be executed in a different order than shown herein. The various steps shown in the flowcharts may be implemented in any way by any circuit structure and/or tangible mechanism (such as software, hardware (such as logic functions implemented by processors or chips) running on a computer device, and/or any combinations thereof).

[0148] Those skilled in the art can understand that in the above methods of specific embodiments, the describing order of the respective steps do not imply a strict execution order, which then constitutes any limitation on the implementation process. The specific execution order of the respective steps should be determined by their functions and possible internal logics.

[0149] Based on the same inventive concept, embodiments of the present application also provide data processing apparatuses corresponding to the data processing methods. As the principle of solving the problem of the apparatuses in the embodiments of the present application is similar to that of the data processing methods of the embodiments of the present application, the implementation of the apparatuses may refer to the implementation of the method, and the repetition will not be repeated.

[0150] Referring to FIG. **6**, it is a schematic diagram of a data processing apparatus provided according to an embodiment of the present application, the apparatus comprising: an acquisition module **601**, a decomposition module **602**, and a sending module **603**; wherein, [0151] the acquisition module **601**, for acquiring label residual values determined for respective data samples, each label residual value being used for representing a degree of deviation between a real label of a corresponding data sample and a label predicted by a target model; [0152] the decomposition module **602**, for performing residual decomposition on the respective data samples based on the respective label residual values to obtain model parameter correction amounts corresponding to the respective data samples; [0153] the sending module **603**, for sending the model parameter correction amounts to a data receiver, so that the data receiver reconstructs model parameter information of the target model based on the model parameter correction amounts.

[0154] By using the above data processing apparatus, on the condition that the label residual values determined for the respective data samples are obtained, residual decomposition may be performed on the respective data samples based on the respective label residual values. Then, the model parameter correction amounts corresponding to the respective data samples obtained from residual decomposition may be sent to the data receiver, and the data receiver may reconstruct the model parameter information based on the model parameter correction amounts. In the present application, the data sender (i.e., the labeled party) can effectively perturb the residuals during the residual decomposition process, so that the data receiver (i.e., the unlabeled party) can only construct the model with erroneous label information. This enables the data sender to protect the real labels owned thereby, and at the same time, the labeled party can transmit necessary correction information (i.e., model parameter correction amounts) to the unlabeled party, allowing the unlabeled party to reconstruct accurate model parameter information based on the correction information so as to ensure the model training performance.

[0155] In a possible embodiment, the decomposition module **602** is used to perform residual decomposition on the respective data samples based on the respective label residual values according to the following steps, and obtain model parameter correction amounts corresponding to the respective data samples, which comprises: [0156] ordering the respective label residual values to obtain respective ordered label residual values; [0157] changing the ordered label residual values where the target data sample is in on the condition that a target data sample that requires residual decomposition is selected from the respective data samples, to obtain residual change information corresponding to the respective data samples; and [0158] determining model parameter correction amounts corresponding to the respective data samples based on the residual change information corresponding to the respective data samples and the real labels of the respective data samples.

[0159] In a possible embodiment, the decomposition module **602** is used to select the target data sample that requires residual decomposition from the respective data samples according to the following steps: [0160] grouping the respective ordered label residual values according to a preset number of groups to obtain grouped label residual values; [0161] selecting a preset number of target label residual values from each grouped label residual values; and [0162] determining a data sample corresponding to the target label residual values as the target data sample.

[0163] In a possible embodiment, on the condition that the residual change information corresponding to the respective data samples corresponds to the residual change vector, the decomposition module **602** is used to change the ordered label residual values where the target data sample is in according to the following steps to obtain the residual change information corresponding to the respective data samples: [0164] for each of the data samples, on the condition that it is judged that the data sample is not the target data sample, determining that the data sample corresponds to a first residual change value; or, [0165] on the condition that it is judged that the data sample is the target data sample and the label residual value of the data sample is greater than zero, determining that the data sample corresponds to a second residual change value; or, [0166] on the condition that it is judged that the data sample is the target data sample and the label residual value of the data sample is less than zero, determining that the data sample corresponds to a third residual change value; and [0167] collecting residual change values respectively corresponding to the respective data samples to determine the residual change vector; wherein, the residual change vector is used to represent whether the label residual values of the respective data samples have changed.

[0168] In a possible embodiment, the decomposition module **602** is used to determine the model parameter correction amounts corresponding to the respective data samples based on the residual change information corresponding to the respective data samples and the real labels of the respective data samples according to the following steps: [0169] performing point multiplication operation on the residual change vector and a transpose result of the residual change vector to determine a first operator; and, determining a second operator corresponding to the label values on the condition that the label values corresponding to the real labels of the respective data samples are determined; and [0170] determining the model parameter correction amounts corresponding to the respective data samples based on the first operator and the second operator.

[0171] In a possible embodiment, the sending module **603** is also used for: [0172] determining changed label residual values based on the residual change information corresponding to the respective data samples and product operation between the respective ordered label residual values after obtaining the residual change information corresponding to the respective data samples; [0173] determining the model parameter information to be sent to the data receiver based on the changed label residual values; and [0174] sending the model parameter information to the data receiver.

[0175] In a possible embodiment, the sending module **603** is used to send the model parameter information to the data receiver according to the following step: [0176] sending the model parameter information to the data receiver on the condition that model convergence conditions are not reached; [0177] the sending module **603** is used to send the model parameter correction amounts to the data receiver according to the following step: [0178] sending the model parameter correction amounts to the data receiver on the condition that the model convergence conditions are reached.

[0179] Referring to FIG. **7**, it is a schematic diagram of another data processing apparatus provided according to an embodiment of the present application, the apparatus comprising: a receiving module **701** and a reconstruction module **702**; wherein, [0180] the receiving module **701** is used to receive model parameter correction amounts corresponding to respective data samples sent by a data sender; wherein, the model parameter correction amounts are determined by means of residual decomposition based on label residual values determined for the respective data samples, each label residual value being used for representing a degree of deviation between a real label of a corresponding data sample and a label predicted by a target model; and [0181] the reconstruction module **702** is used to reconstruct model parameter information of the target model based on the model parameter correction amounts.

[0182] In a possible embodiment, the reconstruction module **702** is used to reconstruct the model parameter information of the target model based on the model parameter correction amounts according to the following steps: [0183] acquiring data characteristic information input for the target model; and [0184] determining the model parameter information of the target model based on the data characteristic information and the model parameter correction amounts.

[0185] In a possible embodiment, on the condition that the data characteristic information comprises a data characteristic vector, the reconstruction module **702** is used to determine the model parameter information of the target model based on the data characteristic information and the model parameter correction amounts according to the following steps: [0186] performing point multiplication operation on the data characteristic vector and a transpose result of the data characteristic vector to determine a third operator; and, performing point multiplication operation on the transpose result of the data characteristic vector and the data characteristic vector to determine a fourth operator; and [0187] determining the model parameter information of the target model based on the third operator, the fourth operator, and the model parameter correction amounts.

[0188] It should be noted that the apparatuses in the embodiments of the present application may implement various processes of the embodiments of the aforementioned method, and achieve the same effects and functions, which will not be repeated here.

[0189] An embodiment of the present application also provides an electronic device. As shown in FIG. **8**, it is a structural schematic diagram of an electronic device provided according to an embodiment of the present application, comprising: a processor **801**, a memory **802**, and a bus **803**. The memory **802** stores machine-readable instructions executable by the processor **801** (such as the execution instructions corresponding to the acquisition module **601**, decomposition module **602**, and sending module **603** in the apparatus shown in FIG. **6**; or such as the execution instructions corresponding to the receiving module **701** and reconstruction module **702** in the apparatus shown in FIG. **7**, and so on). When the electronic device is running, the processor **801** and the memory **802** communicate with each other through the bus **803**, and the machine-readable instructions are executed by the processor **801** to execute the steps of the data processing method shown in FIG. **1** or FIG. **5**.

[0190] An embodiment of the present application also provides a computer-readable storage medium with a computer program stored thereon. The computer program is executed by a processor to execute the steps of the data processing method described in the above embodiments of the method. Wherein, the storage medium may be a volatile or non-volatile computer-readable storage medium.

[0191] An embodiment of the present application also provides a computer program product carrying program code, wherein the instructions included in the program code may be used to execute the steps of the data processing method described in the above embodiments of the method. Please refer to the above embodiments of the method for details, which will not be repeated here.

[0192] Wherein, the above-mentioned computer program product may be specifically implemented through hardware, software, or a combination thereof. In an optional embodiment, the computer program product is embodied as a computer storage medium, while in another optional embodiment, the computer program product is embodied as a software product, such as a Software Development Kit (SDK), and the like.

[0193] The various embodiments in the present application are described in a progressive manner. The same and similar parts between the various embodiments may be referred to each other. Each embodiment focuses on the differences from other embodiments. In particular, for the embodiments of apparatus, device, and computer-readable storage medium, as they are basically similar to those of the method, the description therefor is simplified. For relevant information, please refer to the part of description for the embodiments of the method.

[0194] The apparatus, device, and computer-readable storage medium provided in the embodiments of the present application correspond to the method on a one-on-one basis. Therefore, the apparatus, device, and computer-readable storage medium also have beneficial technical effects similar to those of the method corresponding to them. As the beneficial technical effects of the method have been described in detail above, the beneficial technical effects of the apparatus, device, and computer-readable storage medium will not be repeated here.

[0195] Those skilled in the art should understand that the embodiments of the present application may be provided as method, apparatus (device or system), or computer-readable storage medium. Therefore, the present application may take the form of a fully hardware implementation, a fully software implementation, or a combination of software and hardware implementations. In addition, the present application may take the form of computer-readable storage medium implemented on one or more computer-usable storage medium (including but not limited to disk storage, CD-ROM, optical storage, etc.) containing computer-usable program code.

[0196] The present application is described with reference to the flowcharts and/or block diagrams of the method, apparatus (device or system), and computer-readable storage medium according to the embodiments of the present application. It should be understood that each process and/or block in the flowcharts and/or block diagrams, as well as the combination(s) of processes and/or blocks in the flowcharts and/or block diagrams, may be implemented by computer program instructions. These computer program instructions may be provided to a processor of a general-purpose computer, special-purpose computer, embedded processor, or other programmable data processing apparatus to generate a machine, such that the instructions executed by the processor of the computer or other programmable data processing apparatus generate an apparatus for implementing the functions specified in one or more processes in the flowcharts and/or one or more blocks in the block diagrams.

[0197] These computer program instructions may also be stored in computer-readable memory that may guide a computer or other programmable data processing apparatus to operate in a specific manner, causing the instructions stored in the computer-readable memory to produce a manufactured product including an instruction apparatus that implements the functions specified in one or more processes in the flowcharts and/or one or more blocks in the block diagrams.

[0198] These computer program instructions may also be loaded onto a computer or other programmable data processing apparatus, enabling a series of operation steps to be executed on the computer or other programmable device to generate computer implemented processing. As a result, the instructions executed on the computer or other programmable device provide steps for implementing the functions specified in one or more processes in the flowcharts and/or one or more blocks in the block diagrams.

[0199] In a typical configuration, a computing device includes one or more processors (CPUs), input/output interfaces, network interfaces, and memory.

[0200] The memory may include non-permanent memory, random access memory (RAM) and/or non-volatile memory in computer-readable media, such as read-only memory (ROM) or flash RAM. Memory is an example of computer-readable media.

[0201] Computer-readable media including permanent and non-permanent, removable and non-removable media may be implemented by any method or technology for information storage. Information may be computer-readable instructions, data structures, modules of a program, or other data. Examples of computer storage media include, but are not limited to, phase change memory (PRAM), static random access memory (SRAM), dynamic random access memory (DRAM), other types of random access memory (RAM), read-only memory (ROM), electrically erasable programmable read-only memory (EEPROM), flash memory or other memory technologies, compact disc read-only memory (CD-ROM), digital versatile disc (DVD) or other optical storage, magnetic tape cartridges, magnetic disk storage or other magnetic storage devices, or any other non-transmitting medium that may be used to store information that may be accessed by computing devices. In addition, although the operations of the method of the present application are described in a specific order in the accompanying drawings, this does not require or imply that these operations need be performed in that specific order, or that all shown operations need be performed so as to achieve the desired results. Additionally or alternatively, certain steps may be omitted, multiple steps may be merged into one step for execution, and/or one step may be decomposed into multiple steps for execution.

[0202] Although the spirit and principle of the present application have been described with reference to several specific embodiments, it should be understood that the present application is not limited to the specific embodiments disclosed herein. The division of various aspects does not mean that the features in these aspects cannot be combined for benefit. This division is only for the convenience of expression. The present application is intended to cover various modifications and equivalent arrangements included within the spirit and scope of the appended claims.

Claims

1. A data processing method, comprising: acquiring label residual values determined for respective data samples, each label residual value being used for representing a degree of deviation between a real label of a corresponding data sample and a label predicted by a target model; performing residual decomposition on the respective data samples based on respective label residual values to obtain model parameter correction amounts corresponding to the respective data samples; and sending the model parameter correction amounts to a data receiver, so that the data receiver reconstructs model parameter information of the target model based on the model parameter correction amounts; wherein, performing residual decomposition on the respective data samples based on respective label residual values to obtain model parameter correction amounts corresponding to the respective data samples, comprises: ordering the respective label residual values in an order of data size to obtain respective ordered label residual values; changing the ordered label residual values where the target data sample is in on the condition that a target data sample that requires residual decomposition is selected from the respective data samples, to obtain residual change information corresponding to the respective data samples; and determining model parameter correction amounts corresponding to the respective data samples based on the residual change information corresponding to the respective data samples and the real labels of the respective data samples.

2. (canceled)

3. The method according to claim 1, wherein, the target data sample that requires residual decomposition is selected from the respective data samples according to following steps: grouping the respective ordered label residual values according to a preset number of groups to obtain grouped label residual values; selecting a preset number of target label residual values from each grouped label residual values; and determining a data sample corresponding to the target label residual values as the target data sample.

4. The method according to claim **2** or **31**, wherein, changing the ordered label residual values where the target data sample is in on the condition that the residual change information corresponding to the respective data samples corresponds to a residual change vector, to obtain the residual change information corresponding to the respective data samples, comprises: for each of the data samples, on the condition that it is judged that the data sample is not the target data sample, determining that the data sample corresponds to a first residual change value; or, on the condition that it is judged that the data sample is the target data sample and the label residual value of the data sample is greater than zero, determining that the data sample corresponds to a second residual change value; or, on the condition that it is judged that the data sample is the target data sample and the label residual value of the data sample is less than zero, determining that the data sample corresponds to a third residual change value; and collecting residual change values respectively corresponding to the respective data samples to determine the residual change vector; wherein, the residual change vector is used to represent whether the label residual values of the respective data samples have changed.

5. The method according to claim 3, wherein, determining model parameter correction amounts corresponding to the respective data samples based on the residual change information corresponding to the respective data samples and the real labels of the respective data samples, comprises: performing point multiplication operation on the residual change vector and a transpose result of the residual change vector to determine a first operator; and, determining a second operator corresponding to the label values on the condition that the label values corresponding to the real labels of the respective data samples are determined; and determining the model parameter correction amounts corresponding to the respective data samples based on the first operator and the second operator.

6. The method according to claim 1, wherein, after obtaining the residual change information corresponding to the respective data samples, the method further comprises: determining changed label residual values based on the residual change information corresponding to the respective data samples and product operation between the respective ordered label residual values; determining the model parameter information to be sent to the data receiver based on the changed label residual values; and sending the model parameter information to the data receiver.

7. The method according to claim 5, wherein, sending the model parameter information to the data receiver comprises: sending the model parameter information to the data receiver on the condition that model convergence conditions are not reached, sending the model parameter correction amounts to the data receiver comprises: sending the model parameter correction amounts to the data receiver on the condition that the model convergence conditions are reached.

8-14. (canceled)