

MODEL GRADIENT DETERMINING METHODS, APPARATUSES, DEVICES, AND MEDIA BASED ON FEDERATED LEARNING

TECHNICAL FIELD

[0001] This application relates to the field of computer technologies, and in particular, to model gradient determining methods, apparatuses, devices, and media based on federated learning.

BACKGROUND

[0002] As the society becomes more and more aware of the necessity of data privacy, emerging laws such as the General Data Protection Regulation (GDPR) impose strict restrictions on data circulation, leading to an increasingly severe problem of data islands. As an emerging privacy protection distributed machine learning method, federated learning provides a new idea for alleviating the problem of data islands. However, a plurality of data owners participate in current federated learning training to obtain a unified training model so that models obtained by participants are the same, and it is unable to provide participants with different models based on different scenarios.

[0003] Therefore, how to provide different models for different participants in federated learning is an urgent technical problem to be alleviated.

SUMMARY

[0004] Embodiments of this specification provide model gradient determining methods, apparatuses, devices, and media based on federated learning, to alleviate a

problem that models obtained by participants in existing federated learning are consistent.

[0005] To alleviate the previous technical problem, the embodiments of this specification are implemented as follows:

[0006] Some embodiments of this specification provide a model gradient determining method based on federated learning, including: obtaining data volume information of a participating node, where the data volume information is used to indicate an amount of data used by the participating node to train a basic training model based on local data, and the local data includes user data of a target organization corresponding to the participating node; obtaining a node local gradient obtained by training the basic training model based on the local data by the participating node; determining, based on the data volume information and the node local gradient, a global gradient of a federated learning model that the participating node participates in; determining a degree of participation of the participating node based on the node local gradient of the participating node and the global gradient, where the degree of participation is used to indicate a degree of participation of the participating node in federated learning model training; and determining an actual model gradient of the participating node based on the degree of participation.

[0007] Some embodiments of this specification provide a model gradient determining apparatus based on federated learning, including: a volume information acquisition module, configured to obtain data volume information of a participating node, where the data volume information is used to indicate an amount of data used by the participating node to train a basic training model based on local data, and the local data includes user data of a target organization corresponding to the participating node; a local gradient acquisition module, configured to obtain a node local gradient obtained by training the basic training model based on the local data by the participating node; a global gradient determining module, configured to determine, based on the data volume information and the node local gradient, a global gradient of a federated learning model that the participating node participates in; a degree-of-participation determining module, configured to determine a degree of participation of the participating node based on the

node local gradient of the participating node and the global gradient, where the degree of participation is used to indicate a degree of participation of the participating node in federated learning model training; and an actual gradient determining module, configured to determine an actual model gradient of the participating node based on the degree of participation.

[0008] Some embodiments of this specification provide a model gradient determining device based on federated learning, including: at least one processor; and a memory communicatively connected to the at least one processor, where the memory stores an instruction that can be executed by the at least one processor, and the instruction is executed by the at least one processor so that the at least one processor can: obtain data volume information of a participating node, where the data volume information is used to indicate an amount of data used by the participating node to train a basic training model based on local data, and the local data includes user data of a target organization corresponding to the participating node; obtain a node local gradient obtained by training the basic training model based on the local data by the participating node; determine, based on the data volume information and the node local gradient, a global gradient of a federated learning model that the participating node participates in; determine a degree of participation of the participating node based on the node local gradient of the participating node and the global gradient, where the degree of participation is used to indicate a degree of participation of the participating node in federated learning model training; and determine an actual model gradient of the participating node based on the degree of participation.

[0009] Some embodiments of this specification provide a computer-readable medium, storing a computer-readable instruction, where when the computer-readable instruction can be executed by a processor to implement a model gradient determining method based on federated learning.

[0010] Embodiments of this specification can achieve the following beneficial effects: in the embodiments of this specification, when federated learning is performed, the data volume information used by the participating node to train the basic training model based on the local data as well as the node local gradient provided by the

participating node can be obtained, the global gradient of the federated learning model can be determined, the degree of participation of the participating node can be determined based on the node local gradient and the global gradient, and then the actual model gradient available to the participating node can be determined based on the degree of participation. In this way, different model gradients can be assigned to different participating nodes in the federated learning process so that different federated learning models can be obtained for different participating nodes.

BRIEF DESCRIPTION OF DRAWINGS

[0011] To describe the technical solutions in the embodiments of this specification or in the existing technology more clearly, the following briefly describes the accompanying drawings needed for describing the embodiments or the existing technology. Clearly, the accompanying drawings in the following description merely show some embodiments of this application, and a person of ordinary skill in the art can still derive other drawings from these accompanying drawings without creative efforts.

[0012] FIG. 1 is a schematic diagram illustrating an application scenario of a model gradient determining method based on federated learning according to some embodiments of this specification;

[0013] FIG. 2 is a schematic flowchart illustrating a model gradient determining method based on federated learning according to some embodiments of this specification;

[0014] FIG. 3 is a swimlane diagram illustrating a model gradient determining method based on federated learning according to some embodiments of this specification;

[0015] FIG. 4 is a schematic structural diagram of a model gradient determining apparatus based on federated learning according to some embodiments of this specification; and

[0016] FIG. 5 is a schematic structural diagram of a model gradient determining

device based on federated learning according to some embodiments of this specification.

DESCRIPTION OF EMBODIMENTS

[0017] To make the objectives, technical solutions, and advantages of one or more embodiments of this specification clearer, the following clearly and comprehensively describes the technical solutions of the one or more embodiments of this specification with reference to specific embodiments and accompanying drawings of this specification. Clearly, the described embodiments are merely some rather than all of the embodiments of this specification. Other embodiments obtained by a person of ordinary skill in the art based on the embodiments of this specification without creative efforts shall fall within the protection scope of one or more embodiments of this specification.

[0018] The following describes in detail the technical solutions in the embodiments of this specification with reference to the accompanying drawings.

[0019] In the existing technology, federated learning usually needs each participant to obtain a model from a server; then each participant uses local data to train the model, and uploads obtained model gradient data to the server. The server aggregates each piece of gradient data to obtain an updated model parameter or gradient, and then sends the updated model parameter or gradient to each participant so that each participant can obtain a unified model obtained through federated learning.

[0020] To alleviate the defects in the existing technology, the technical solutions of this specification provide the following embodiments:

[0021] FIG. 1 is a schematic diagram illustrating an application scenario of a model gradient determining method based on federated learning according to some embodiments of this specification. As shown in FIG. 1, the solutions can include the following: participating nodes 1 and a server 2 participating in federated learning. The participating nodes 1 participating in the federated learning can include a plurality of participating nodes, and the server 2 can obtain model gradients provided by the plurality of participating nodes, and then perform aggregation to obtain a federated learning model. In the embodiments of this specification, the participating nodes 2 can

train a basic training model based on local data of the participating nodes, and feed back model parameters of the trained model or node local gradients to the server; and the server 2 can further obtain data volume information of training data collected by the participating nodes during model training, and obtain a global gradient of the federated learning model based on the data volume information of the participating nodes and the model parameters or the node local gradients provided by the participating nodes. In the embodiments of this specification, the server 2 can further determine degrees of participation of the participating nodes in the federated learning based on the node local gradients of the participating nodes 1 and the global gradient, and then determine actual model gradients matching the participating nodes based on the degrees of participation.

[0022] Next, a model gradient determining method based on federated learning provided in some embodiments of this specification is described in detail with reference to the accompanying drawings:

[0023] FIG. 2 is a schematic flowchart illustrating a model gradient determining method based on federated learning according to some embodiments of this specification. From a program perspective, the procedure can be executed by a program or an application client mounted on an application server.

[0024] As shown in FIG. 2, the procedure can include the following steps:

[0025] Step 202: Obtain data volume information of a participating node, where the data volume information is used to indicate an amount of data used by the participating node to train a basic training model based on local data, and the local data includes user data of a target organization corresponding to the participating node.

[0026] Step 204: Obtain a node local gradient obtained by training the basic training model based on the local data by the participating node.

[0027] In the embodiments of this specification, a participating node can be any one of a plurality of participating nodes participating in federated learning. The participating node can represent various organizations participating in the federated learning, including organizations that provide services such as resource processing, payment, leasing, and online and offline transactions. The user data can include user data retained by a user in the target organization. For example, the user data can include service data

and transaction records of the user handling services in the target organization, can also include registration information provided by the user to the target organization, and can also include browsing records of the user in the target organization.

[0028] The participating node can train the basic training model based on data obtained by the participating node, to obtain the node local gradient, and send the node local gradient and the data volume information participating in the training to the server. The data volume information can include an amount of data indicating training data used for model training, and the training data does not need to be sent to the server. It can be ensured that data of the participating node does not leave the domain, and security of the data can be improved.

[0029] The basic training model can represent the latest training model existing in the participating node. In practice, the basic training model can be a model to be trained provided by the server to the participating node. The federated learning can use a plurality of rounds of iterative learning. For the first round of training, the server can send the basic training model to each participating node; after the participating node uses local data to train the model, the trained model or a model parameter can be temporarily stored. In subsequent rounds of training, a model obtained after previous rounds can be updated and trained. In this case, the basic training model can represent the model obtained after the previous rounds of training.

[0030] Step 206: Determine, based on the data volume information and the node local gradient, a global gradient of a federated learning model that the participating node participates in.

[0031] In practice, in the federated learning, the global gradient is usually obtained just by aggregating the node local gradient provided by the participating node. In the embodiments of this specification, the global gradient of the federated learning model can be obtained with reference to the data volume information of the participating node. Accuracy of the global gradient can be improved.

[0032] Step 208: Determine a degree of participation of the participating node based on the node local gradient of the participating node and the global gradient, where the degree of participation is used to indicate a degree of participation of the

participating node in federated learning model training.

[0033] In practice, due to the impacts of a computing capability of a participating node or quality or an amount of training data, different participating nodes may have different degrees of participation in the federated learning process. For example, the node local gradient provided by the participating node that uses training data with better quality and a larger data volume for model training has a greater influence on the global gradient, and the participating node can have a higher degree of participation.

[0034] Step 210: Determine an actual model gradient of the participating node based on the degree of participation.

[0035] The server can send the determined actual model gradient of the participating node to a corresponding participating node so that the participating node can obtain a federated learning model that matches the participating node. In practice, the server can further determine the federated learning model that matches the participating node or a model parameter based on the determined actual model gradient of the participating node, and can also send the federated learning model or the model parameter to the participating node. The federated learning model can include a model for evaluation, for example, a risk evaluation model, a reputation evaluation model, or a profit and loss evaluation model.

[0036] In the embodiments of this specification, when federated learning is performed, the data volume information used by the participating node to train the basic training model based on the local data as well as the node local gradient provided by the participating node can be obtained, the global gradient of the federated learning model can be determined, the degree of participation of the participating node can be determined based on the node local gradient and the global gradient, and then the actual model gradient available to the participating node can be determined based on the degree of participation. In this way, different model gradients can be assigned to different participating nodes in the federated learning process so that different federated learning models can be obtained for different participating nodes.

[0037] It should be understood that in the methods described in one or more embodiments of this specification, the order of some steps can be exchanged based on

actual needs, or some steps can be omitted or deleted.

[0038] Based on the method in FIG. 2, the embodiments of this specification further provide some specific implementations of the method, which are described below.

[0039] To improve accuracy of the global gradient in the federated learning, data quality of participating nodes can also be incorporated into determining of the global gradient in the embodiments of this specification. Optionally, in the embodiments of this specification, after the obtaining a node local gradient obtained by training the basic training model based on the local data by the participating node, the method can further include the following: obtaining a marginal loss of the participating node, where the marginal loss is used to represent a degree of influence of the node local gradient of the participating node on performance of the federated learning model; and determining node mass of the participating node based on the marginal loss; and the determining, based on the data volume information and the node local gradient, a global gradient of a federated learning model that the participating node participates in can specifically include the following: determining, based on the data volume information, the node local gradient, and the node mass, the global gradient of the federated learning model that the participating node participates in.

[0040] The marginal loss can represent a loss of the federated learning model that increases with an increase of participating nodes in the federated learning, and can represent a degree of influence of a node local gradient of a participating node on performance of the federated learning model.

[0041] In the embodiments of this specification, the participating node participating in the federated learning model can include a plurality of participating nodes. The obtaining a marginal loss of the participating node can specifically include the following: determining a first reference global model based on a node local gradient of each participating node in the plurality of participating nodes; determining a second reference global model based on a node local gradient of each participating node other than the participating node in the plurality of participating nodes; determining a first model loss of the first reference global model based on a predetermined verification set; determining a second model loss of the second reference global model based on the

predetermined verification set; and determining the marginal loss of the participating node based on the first model loss and the second model loss.

[0042] As an implementation, assume that in each round t of the federated learning, $M_G^{(t)}$ is used to represent a first global model obtained by aggregating the node local gradients provided by the participating nodes, and $M_{-i}^{(t)}$ is used to represent a second global model obtained by aggregating node local gradients of participating nodes except a participating node i ; and $l^{(t)}$ and $l_{-i}^{(t)}$ respectively represent losses of the models $M_G^{(t)}$ and $M_{-i}^{(t)}$ on the validation set. Then, a marginal loss $\delta_i^{(t)}$ can be $\delta_i^{(t)} = l_{-i}^{(t)} - l^{(t)}$.

[0043] Here, $M_G^{(t)} = M_G^{(t-1)} + \frac{1}{N} \sum_{j \in W} u_j^{(t)}$; $M_{-i}^{(t)} = M_G^{(t-1)} + \frac{1}{N-1} \sum_{j \in w'} u_j^{(t)}$;

$M_G^{(t-1)}$ represents a federated learning model in a $(t-1)^{\text{th}}$ round of the federated learning;

N represents a total quantity of participating nodes participating in the federated learning; W represents a set of the participating nodes participating in the federated learning, and j represents any participating node participating in the federated learning;

$\sum_{j \in W} u_j^{(t)}$ represents the sum of node local gradients of participating nodes in the

federated learning; w' represents a set of participating nodes except the participating

node i among participating nodes participating in the federated learning; and $\sum_{j \in w'} u_j^{(t)}$

represents the sum of node local gradients of participating nodes other than participating node i .

[0044] In practice, the first model loss $l^{(t)}$ of the first reference global model $M_G^{(t)}$ and the second model loss $l_{-i}^{(t)}$ of the second reference global model $M_{-i}^{(t)}$ can be calculated by using the predetermined verification set based on a loss function.

[0045] In the embodiments of this specification, a larger marginal loss $\delta_i^{(t)}$ of the participating node i indicates a more important local gradient of the participating node

i in the federated learning. Optionally, in the embodiments of this specification, the determining node mass of the participating node based on the marginal loss can specifically include the following: determining the node mass of the participating node based on a marginal loss of each participating node in the plurality of participating nodes and a normalization algorithm.

[0046] In the embodiments of this specification, node mass $m_i^{(t)}$ of the participating node i in the t^{th} round of the federated learning training can be expressed as $m_i^{(t)} = \frac{w_0^{(t)} + w_i^{(t)}}{\sum_i (w_0^{(t)} + w_i^{(t)})}$; $w_0^{(t)}$ represents a constant in the t^{th} round, and $w_i^{(t)} = \frac{\delta_i^{(t)} - \min_i(\delta_i^{(t)})}{\sum_i (\delta_i^{(t)} - \min_i(\delta_i^{(t)}))}$; $\min_i(\delta_i^{(t)})$ represents the smallest marginal loss in marginal losses of the participating nodes in the t^{th} round of the training; and $\sum_i (\delta_i^{(t)} - \min_i(\delta_i^{(t)}))$ represents the sum of a difference between a marginal loss of each participating node and the smallest marginal loss in the t^{th} round of the training.

[0047] The node mass $m_i^{(t)}$ can also represent a weight of the participating node i in the federated learning, and a local gradient with a larger marginal loss value can correspond to a greater weight. $w_0^{(t)}$ can be a positive number, and it can ensure that the node mass $m_i^{(t)}$ is not equal to 0.

[0048] To obtain a more effective global gradient, in the embodiments of this specification, effective participating nodes can also be determined based on marginal losses, and the global gradient can be determined based on node local gradients of the effective participating nodes. The embodiments can also make the aggregated model converge faster and have higher accuracy.

[0049] Optionally, the method in the embodiments of this specification can further include the following: determining a participating node with a marginal loss greater than or equal to a predetermined loss threshold in the plurality of participating nodes as an effective participating node; and the determining, based on the data volume information, the node local gradient, and the node mass, the global gradient of the federated learning model that the participating node participates in specifically includes

the following: performing an aggregation operation on a node local gradient of each of the effective participating node based on data volume information of the participating node and node mass of the participating node, to obtain the global gradient.

[0050] In the embodiments of this specification, a global gradient $u_G^{(t)}$ in the t^{th} round of the training can be obtained based on the following equation:

[0051] $u_G^{(t)} = \frac{\sum_i m_i^{(t)} |D_i| u_i^{(t)}}{\sum_i m_i^{(t)} |D_i|}$; i represents an effective participating node with a marginal loss greater than or equal to the predetermined loss threshold in the participating node participating in the federated learning; D_i represents a data volume, of the participating node i , of training data used for training based on local data; and $\sum_i$ represents the sum of related data for effective participating nodes.

[0052] To improve effectiveness of data, node mass of the effective participating node can be calculated when calculating the node mass of the participating node in the embodiments of this specification, where node mass of each effective participating node is determined based on a marginal loss of each effective participating node and the normalization algorithm.

[0053] In the above equation $m_i^{(t)} = \frac{w_0^{(t)} + w_i^{(t)}}{\sum_i (w_0^{(t)} + w_i^{(t)})}$, i can represent any node in effective participating nodes, $\min_i(\delta_i^{(t)})$ can represent the smallest marginal loss in marginal losses of the effective participating nodes in the t^{th} round of the training; and $\sum_i(\delta_i^{(t)} - \min_i(\delta_i^{(t)}))$ represents the sum of a difference between a marginal loss of each effective participating node and the smallest marginal loss in the t^{th} round of the training.

[0054] In the embodiments of this specification, the participating node participating in the federated learning model include a plurality of participating nodes, and the degree of participation of the participating node can be represented by a contribution degree of the participating node in the federated learning. The determining a degree of participation of the participating node based on the node local gradient of the participating node and the global gradient can specifically include the following: determining a node contribution degree of each of the plurality of participating nodes

based on a node local gradient of the participating node and the global gradient; and determining a relative contribution degree of the participating node based on a node contribution degree of the participating node and the node contribution degree of each participating node.

[0055] In the embodiments of this specification, distance mapping of the node local gradients of the participating nodes to the global gradient can be used as a benefit function of contribution evaluation, and then the contribution degrees of the participating nodes can be calculated by using Shapley values. Specifically, in the t^{th} round of the training, a benefit function $V(i)$ of the participating node i can be: $V(i) = |u_i^{(t)}| \cos(u_i^{(t)}, u_G^{(t)})$. Then, the previously described benefit function can be substituted into calculation using the Shapley values, to obtain a node contribution degree $\phi_i^{(t)}: \phi_i^{(t)} = \alpha_i^{(t)} |u_i^{(t)}| \cos(u_i^{(t)}, u_G^{(t)})$ of the participating node i in the t^{th} round, where $\alpha_i^{(t)}$ represents an aggregation weight of the participating node i in the t^{th} round.

[0056] In the embodiments of this specification, the Shapley values are used to calculate the contribution degrees of the participating nodes, and the distance mapping of the node local gradients of the participating nodes on the global gradient is used as the benefit function of the Shapley values. The method satisfies the idea of contribution evaluation in cooperative games, overcomes the disadvantage of using only accuracy in related work as a contribution, and has smaller computational overheads than those of real Shapley values, thereby saving resources and improving computational efficiency.

[0057] To better reflect stability of the participating node in the entire federated learning, in the embodiments of this specification, the relative contribution degree of the participating node in the entire training process can be further determined based on the contribution degree of the participating node in each round of training. A node cumulative contribution of each participating node can be determined based on a node contribution degree of each participating node; and a relative contribution degree of the participating node is determined based on the node cumulative contribution of the participating node and the largest node cumulative contribution in node cumulative

contributions of participating nodes.

[0058] As an implementation, a relative contribution degree $z_i^{(t)}$ of the participating node i in the t^{th} round of the training can be expressed as:

[0059] $z_i^{(t)} = \frac{c_i^{(t)}}{\max_i(c_i^{(t)})}$, where

[0060] $c_i^{(t)}$ represents a cumulative contribution of the participant i from the 1st round to the t^{th} round t , and $c_i^{(t)} = \max(0, \sum_{i=1}^t \phi_i^{(t)})$.

[0061] In the embodiments of this specification, the actual model gradient of the participating node can be determined based on the relative contribution degree. A higher relative contribution degree of the participating node indicates an actual model gradient obtained by the participating node that is closer to the global gradient.

[0062] In the embodiments of this specification, reputations of participating nodes can be further determined based on the historical behavior of the participating nodes, and can also be used to indicate reliability or trustworthiness of the participating nodes. Fluctuations of the participating nodes in the federated learning can be smoothened, and the degrees of participation of the participating nodes can be more accurately determined.

[0063] In the embodiments of this specification, the global gradient includes a gradient obtained through a target quantity of rounds of iterative calculations. The determining a degree of participation of the participating node can further include the following: obtaining a trustworthiness parameter of the participating node, where the trustworthiness parameter is used to represent a comprehensive degree of reliability of the participating node in the target quantity of rounds of calculation processes of the global gradient; and determining a reputation degree of the participating node based on the relative contribution degree and the trustworthiness parameter.

[0064] As an implementation, in the embodiments of this specification, a reputation degree $r_i^{(t)}$ of the participating node i in the t^{th} round based on $r_i^{(t)} = q_i^{(t)} z_i^{(t)}$, where $q_i^{(t)}$ represents a trustworthiness parameter of the participating node i in the t^{th} round,

and $z_i^{(t)}$ represents a relative contribution degree of the participating node i in the t^{th} round.

[0065] In the embodiments of this specification, the obtaining a trustworthiness parameter of the participating node can specifically include the following: determining a first quantity of times that the participating node is determined as an effective participating node in the target quantity of rounds, where the effective participating node represents a participating node with a marginal loss greater than or equal to a predetermined loss threshold; determining a second quantity of times that the participating node is determined as an ineffective participating node in the target quantity of rounds, where the ineffective participating node represents a participating node with a marginal loss less than the predetermined loss threshold; and determining the trustworthiness parameter of the participating node based on the first quantity of times and the second quantity of times.

[0066] As an implementation, in the embodiments of this specification, $q_i^{(t)} = ae^{be^{cx_i^{(t)}}}$ can be used to represent the trustworthiness parameters $q_i^{(t)}$ of the participating nodes, can represent degrees of trustworthiness of the participating nodes in the t^{th} round, and can also represent a comprehensive degree of reliability of the participating nodes in the entire federated learning for calculating the global gradient from the 1st round to the t^{th} round, where $x_i^{(t)} = \frac{\beta n_i^{\text{pass}} - (1-\beta)n_i^{\text{fail}}}{\beta n_i^{\text{pass}} + (1-\beta)n_i^{\text{fail}}}$, β represents a constant coefficient, and can be set based on actual needs, n_i^{pass} represents a quantity of times that the participating node i is determined as an effective participating node in the training process from the 1st round to the t^{th} round, and n_i^{fail} represents a quantity of times that the participating node i is determined as an ineffective participating node in the training process from the 1st round to the t^{th} round.

[0067] In the embodiments of this specification, the global gradient can include a plurality of global gradient factors, the participating node participating in the federated learning model can include a plurality of participating nodes, and the global gradient can include the gradient obtained through the target quantity of rounds of iterative

calculations; and the determining an actual model gradient of the participating node based on the degree of participation can specifically include the following: determining a quantity of matching gradients corresponding to the participating node based on a ratio of the reputation degree of the participating node to the greatest reputation degree, where the greatest reputation degree is used to represent the greatest reputation degree in reputation degrees of the plurality of participating nodes; and selecting global gradient factors of the quantity of matching gradients from the global gradient to obtain the actual model gradient.

[0068] In the embodiments of this specification, at least some of global gradients can be sent to a corresponding participating node based on a reputation degree of the participating node so that the participating node can obtain a model gradient matching the participating node, and each participating node can obtain a federated learning model that matches the node.

[0069] In the embodiments of this specification, a quantity $num_i^{(t)}$ of matching gradients allocated to the participating node i in the t^{th} round can be obtained based on $num_i^{(t)} = \frac{r_i^{(t)}}{\max_i r_i^{(t)}} |u_G^{(t)}|$. Here, $|u_G^{(t)}|$ can represent a total quantity of global gradient factors included in the global gradient in the t^{th} round. $r_i^{(t)} \in (0, 1]$, and a participating node with the highest reputation degree can obtain the entire global gradient. In practice, if the calculated quantity $num_i^{(t)}$ of matching gradients includes decimals, the quantity of matching gradients can also be obtained based on rounding methods such as rounding, rounding up, and rounding down.

[0070] In practice, when participating nodes use local data for model training, since characteristics of the training data used by different participating nodes may have some differences, different participating nodes may have different degrees of influence on model parameters in a global model. Node local gradients provided by the participating nodes can also reflect the needs of the participating nodes. For example, a node local gradient provided by participating node A emphatically reflects a relationship between a user age and user needs for products, and may have a greater influence on the user

age and the user needs for products in the global gradient; a node local gradient provided by participating node B emphatically reflects a relationship between education background of a user and user needs for products, and may have a greater influence on the education background of the user and the user needs for products in the global gradient; and further, gradient factors reflecting a user age and user needs for products in the global gradient are fed back to participating node A, and gradient factors reflecting education background of the user and user needs for products in the global gradient are fed back to participating node B.

[0071] To make an actual gradient allocated to the participating nodes more in line with the needs of the participating nodes, in the embodiments of this specification, the degree of influence of each gradient factor in the node local gradient on the global gradient can also be determined based on the node local gradients provided by the participating nodes. A global gradient factor with a greater influence in the global gradient is sent to the participating node.

[0072] Optionally, in the embodiments of this specification, the global gradient includes a plurality of global gradient factors, and the node local gradient includes a plurality of local gradient factors; and the method in the embodiments of this specification can further include the following: obtaining a node influence degree of each global gradient factor relative to the participating node, where the node influence degree is used to indicate a degree that each global gradient factor is influenced by the participating node; and sorting the global gradient factors in the global gradient based on the node influence degree, to obtain the sorted global gradient factors; and the selecting global gradient factors of the quantity of matching gradients from the global gradient to obtain the actual model gradient can specifically include the following: selecting global gradient factors of the quantity of matching gradients from the sorted global gradient factors based on a predetermined order, to obtain the actual model gradient.

[0073] In practice, global gradient factors in the global gradient can be sorted based on the node influence degree. Optionally, in the embodiments of this specification, the sorting the global gradient factors in the global gradient based on the node influence

degree can specifically include the following: sorting the global gradient factors in the global gradient based on the node influence degree in descending order; and the selecting global gradient factors of the quantity of matching gradients from the sorted global gradient factors, to obtain the actual model gradient specifically includes the following: selecting global gradient factors of the top quantity of matching gradients from the global gradient factors sorted in descending order, to obtain the actual model gradient.

[0074] Assume that the global gradient includes four global gradient factors M, N, P, and Q, node influence degrees of participating node A on these four global gradient factors are A1, A2, A3, and A4, where $A3 > A2 > A4 > A1$, and the quantity of matching gradients corresponding to participating node A is 3. It can be determined that the global gradient factors N, P, and Q with top 3 node influence degrees are sent to the participating node.

[0075] In practice, the global gradient factors in the global gradient can also be sorted in ascending order based on the node influence degrees, and the factors with the last rankings are determined as the actual model gradient of the participating node. The specific sorting method is not limited here, as long as actual needs can be satisfied.

[0076] In the embodiments of this specification, the obtaining a node influence degree of each global gradient factor relative to the participating node can specifically include the following: determining a first distribution parameter corresponding to each local gradient factor in the local gradient of the participating node, where the first distribution parameter is used to indicate a proportion of each local gradient factor in the local gradient; determining a second distribution parameter corresponding to each global gradient factor in the global gradient, where the second parameter is used to indicate a proportion of each global gradient factor in the global gradient; and determining the node influence degree of each global gradient factor relative to the participating node based on the first distribution parameter and the second distribution parameter.

[0077] In practice, the global gradient factors in the global gradient can be sorted in predetermined attribute order, and local gradient factors in the node local gradient

can also be sorted in the predetermined attribute order.

[0078] As an implementation, in the embodiments of this specification, the node influence degree $s_j^{(t)}$ corresponding to the j^{th} node local gradient factor in the node local gradient provided by the participating node i in the t^{th} round, and can be expressed as $s_j^{(t)} = s_{i,j}^{(t)} \times s_{G,j}^{(t)}$, where $s_{i,j}^{(t)}$ represents the first distribution parameter, and $s_{G,j}^{(t)}$ represents the second distribution parameter.

[0079] A first distribution parameter $s_{i,j}^{(t)}$ of the j^{th} node local gradient factor in the node local gradient of the participating node i in the t^{th} round can be expressed as $s_{i,j}^{(t)} = \frac{u_{i,j}^{(t)}}{\sum_j u_{i,j}^{(t)}}$, where $u_{i,j}^{(t)}$ represents the j^{th} node local gradient factor in the node local gradient of the participating node i in the t^{th} round, and $\sum_j u_{i,j}^{(t)}$ represents the sum of node local gradient factors in node local gradients $u_i^{(t)}$ provided by the participating node i in the t^{th} round.

[0080] A second distribution parameter $s_{G,j}^{(t)}$ of the j^{th} global gradient factor in the global gradient in the t^{th} round can be expressed as $s_{G,j}^{(t)} = \frac{u_{G,j}^{(t)}}{\sum_j u_{G,j}^{(t)}}$, where $u_{G,j}^{(t)}$ represents the j^{th} global gradient factor in the global gradient in the t^{th} round, and $\sum_j u_{G,j}^{(t)}$ represents the sum of global gradient factors in the global gradient $u_G^{(t)}$ in the t^{th} round.

[0081] In the embodiments of this specification, the marginal loss is used as a measure of data quality of each participating node in a gradient aggregation process, and gradient aggregation is performed in combination with the data quality and quantity. The embodiments can make the aggregated model converge faster and have higher accuracy.

[0082] Moreover, in the embodiments of this specification, the data quality and contribution of the participating nodes are comprehensively considered, and a cumulative reputation of the participating nodes is calculated to determine the quantity

of gradients allocated to each participating node. When the gradient is selected, the local gradient distribution of the participating nodes is also considered so that the participating nodes can obtain models that match the participating nodes, and participating nodes with higher contributions can obtain higher model accuracy.

[0083] To more clearly illustrate the model gradient determining method based on federated learning provided in the embodiment of this specification, FIG. 3 is a swimlane diagram illustrating a model gradient determining method based on federated learning according to some embodiments of this specification. As shown in FIG. 3, the method can include a node training phase, a gradient aggregation phase, and a gradient determining phase, and can specifically include the following:

[0084] Step 302: A server sends a basic training model to each participating node participating in federated learning.

[0085] Step 304: The participating node obtains the basic training model.

[0086] Step 306: The participating node trains the basic training model based on local data to obtain a trained model.

[0087] Step 308: The participating node can determine a node local gradient based on the trained model.

[0088] Step 310: The participating node sends, to the server, the node local gradient and data volume information of training data used for training the basic training model based on the local data.

[0089] Step 312: The server obtains the node local gradient and the data volume information of the participating node.

[0090] In practice, participating node participating in the federated learning can respectively perform step 304 to step 310, and the server can obtain the node local gradient and the data volume information of each participating node.

[0091] Step 314: After obtaining the node local gradient and the data volume information sent by each participating node participating in the federated learning, the server can determine, based on the data volume information and the node local gradient, the global gradient of the federated learning model that the participating node participates in.

[0092] In the embodiments of this specification, marginal losses of the participating nodes can also be determined based on the node local gradients of the participating nodes, and based on the marginal losses, effective participating nodes and node mass of the participating nodes can be determined from the participating nodes. In the embodiments of this specification, the global gradient of the federated learning model can be determined based on the node local gradients, the data volume information, and node mass of the effective participating nodes, which can effectively improve convergence of the model and improve training efficiency.

[0093] Step 316: The server can determine a contribution degree of the participating node based on the node local gradient and the global gradient, and determine a reputation degree of the participating node based on the contribution degree.

[0094] Step 318: The server can determine a quantity of matching gradients corresponding to the participating node based on the reputation degree of the participating node.

[0095] Step 320: The server determines a node influence degree of each global gradient factor in the global gradient relative to the participating node, sorts the global gradient factors in the global gradient based on the node influence degree, and obtains the sorted global gradient factors.

[0096] Step 322: The server selects global gradient factors of the quantity of matching gradients from the sorted global gradient factors based on a predetermined order, to obtain an actual model gradient.

[0097] Step 324: The server sends the determined actual model gradient matching the participating node to the participating node.

[0098] The server can send the actual model gradient corresponding to each participating node to each participating node, and each participating node can generate a federated learning model based on the actual model gradient that is received by the participating node. For an iterative training process in the federated learning, the participating nodes can update training models of the participating nodes based on the received actual model gradients to obtain the latest version of the training model, where the latest version of the training model can be considered as the basic training model.

Then the participating nodes can train the latest version of the training model based on the data about the participating nodes, and feed back, to the server for aggregation, the node local gradient obtained from the training.

[0099] Based on the same idea, the embodiments of this specification further provide apparatuses corresponding to the previous methods. FIG. 4 is a schematic structural diagram of a model gradient determining apparatus based on federated learning according to some embodiments of this specification. As shown in FIG. 4, the apparatus can include the following: a volume information acquisition module 402, configured to obtain data volume information of a participating node, where the data volume information is used to indicate an amount of data used by the participating node to train a basic training model based on local data, and the local data includes user data of a target organization corresponding to the participating node; a local gradient acquisition module 404, configured to obtain a node local gradient obtained by training the basic training model based on the local data by the participating node; a global gradient determining module 406, configured to determine, based on the data volume information and the node local gradient, a global gradient of a federated learning model that the participating node participates in; a degree-of-participation determining module 408, configured to determine a degree of participation of the participating node based on the node local gradient of the participating node and the global gradient, where the degree of participation is used to indicate a degree of participation of the participating node in federated learning model training; and an actual gradient determining module 410, configured to determine an actual model gradient of the participating node based on the degree of participation.

[0100] Based on the same idea, the embodiments of this specification further provide devices corresponding to the previous methods.

[0101] FIG. 5 is a schematic structural diagram of a model gradient determining device based on federated learning according to some embodiments of this specification. As shown in FIG. 5, the device 500 can include the following: at least one processor 510; and a memory 530 communicatively connected to the at least one processor.

[0102] The memory 530 stores an instruction 520 that can be executed by the at

least one processor 510, and the instruction is executed by the at least one processor 510 so that the at least one processor 510 can: obtain data volume information of a participating node, where the data volume information is used to indicate an amount of data used by the participating node to train a basic training model based on local data, and the local data includes user data of a target organization corresponding to the participating node; obtain a node local gradient obtained by training the basic training model based on the local data by the participating node; determine, based on the data volume information and the node local gradient, a global gradient of a federated learning model that the participating node participates in; determine a degree of participation of the participating node based on the node local gradient of the participating node and the global gradient, where the degree of participation is used to indicate a degree of participation of the participating node in federated learning model training; and determine an actual model gradient of the participating node based on the degree of participation.

[0103] Based on the same idea, the embodiments of this specification further provide a computer-readable medium corresponding to the previous methods. The computer-readable medium stores a computer-readable instruction, where the computer-readable instruction can be executed by a processor to implement the previously described model gradient determining method based on federated learning.

[0104] The embodiments in this specification are described in a progressive way. For same or similar parts of the embodiments, references can be made to the embodiments mutually. Each embodiment focuses on a difference from other embodiments. Particularly, the device shown in FIG. 5 is similar to a method embodiment, and therefore is described briefly. For related parts, references can be made to related descriptions in the method embodiment.

[0105] In the 1990s, whether a technical improvement is a hardware improvement (for example, an improvement to a circuit structure, such as a diode, a transistor, or a switch) or a software improvement (an improvement to a method procedure) can be clearly distinguished. However, as technologies develop, current improvements to many method procedures can be considered as direct improvements to hardware circuit

structures. A designer usually programs an improved method procedure into a hardware circuit, to obtain a corresponding hardware circuit structure. Therefore, a method procedure can be improved by using a hardware entity module. For example, a programmable logic device (PLD) (for example, a field programmable gate array (FPGA)) is such an integrated circuit, and a logical function of the PLD is determined by a user through device programming. The designer performs programming to "integrate" a digital system to a PLD without requesting a chip manufacturer to design and produce an application-specific integrated circuit (ASIC) chip. In addition, at present, instead of manually manufacturing an integrated circuit chip, this type of programming is mostly implemented by using "logic compiler" software. The software is similar to a software compiler used to develop and write a program. Original code needs to be written in a particular programming language for compilation. The language is referred to as a hardware description language (HDL). There are many HDLs, such as the Advanced Boolean Expression Language (ABEL), the Altera Hardware Description Language (AHDL), Confluence, the Cornell University Programming Language (CUPL), HDCal, the Java Hardware Description Language (JHDL), Lava, Lola, MyHDL, PALASM, and the Ruby Hardware Description Language (RHDL). The very-high-speed integrated circuit hardware description language (VHDL) and Verilog are most commonly used. A person skilled in the art should also understand that a hardware circuit that implements a logical method procedure can be readily obtained once the method procedure is logically programmed by using the several described hardware description languages and is programmed into an integrated circuit.

[0106] A controller can be implemented by using any appropriate methods. For example, the controller can be a microprocessor or a processor, or a computer-readable medium that stores computer-readable program code (such as software or firmware) that can be executed by the microprocessor or the processor, a logic gate, a switch, an application-specific integrated circuit (ASIC), a programmable logic controller, or a built-in microprocessor. Examples of the controller include but are not limited to the following microprocessors: ARC 625D, Atmel AT91SAM, Microchip PIC18F26K20, and Silicone Labs C8051F320. The memory controller can also be implemented as a

part of the control logic of the memory. A person skilled in the art also knows that, in addition to implementing the controller by using the computer-readable program code, logic programming can be performed on method steps to allow the controller to implement the same function in forms of the logic gate, the switch, the application-specific integrated circuit, the programmable logic controller, and the built-in microcontroller. Therefore, the controller can be considered as a hardware component, and an apparatus configured to implement various functions in the controller can also be considered as a structure in the hardware component. Or the apparatus configured to implement various functions can even be considered as both a software module implementing the method and a structure in the hardware component.

[0107] The system, apparatus, module, or unit illustrated in the previous embodiments can be implemented by using a computer chip or an entity, or can be implemented by using a product having a certain function. A typical implementation device is a computer. Specifically, the computer can be, for example, a personal computer, a laptop computer, a cellular phone, a camera phone, a smartphone, a personal digital assistant, a media player, a navigation device, an email receiving and sending device, a game console, a tablet computer, a wearable device, or any combination of these devices.

[0108] For ease of description, the apparatus above is described by dividing functions into various units. Certainly, when this application is implemented, a function of each unit can be implemented in one or more pieces of software and/or hardware.

[0109] A person skilled in the art should understand that some embodiments of this application can be provided as a method, a system, or a computer program product. Therefore, this application can use a form of hardware only embodiments, software only embodiments, or embodiments with a combination of software and hardware. Moreover, the embodiments of this application can use a form of a computer program product that is implemented on one or more computer-usable storage media (including but not limited to a disk memory, a CD-ROM, an optical memory, etc.) that include computer-usable program code.

[0110] This application is described with reference to the flowcharts and/or block

diagrams of the method, the device (system), and the computer program product based on the embodiments of this application. It is worthwhile to note that computer program instructions can be used to implement each process and/or each block in the flowcharts and/or the block diagrams and a combination of a process and/or a block in the flowcharts and/or the block diagrams. These computer program instructions can be provided for a general-purpose computer, a dedicated computer, an embedded processor, or a processor of another programmable data processing device to generate a machine so that the instructions executed by the computer or the processor of the another programmable data processing device generate a device for implementing a specific function in one or more processes in the flowcharts and/or in one or more blocks in the block diagrams.

[0111] These computer program instructions can be stored in a computer-readable memory that can instruct the computer or other programmable data processing devices to work in a specific way, so the instructions stored in the computer-readable memory generate an artifact that includes an instruction apparatus. The instruction apparatus implements a specific function in one or more processes in the flowcharts and/or in one or more blocks in the block diagrams.

[0112] These computer program instructions can be loaded onto the computer or another programmable data processing device so that a series of operations and steps are performed on the computer or other programmable devices, thereby generating computer-implemented processing. Therefore, the instructions executed on the computer or other programmable devices provide steps for implementing a specific function in one or more processes in the flowcharts and/or in one or more blocks in the block diagrams.

[0113] In a typical configuration, a computing device includes one or more processors (CPU), an input/output interface, a network interface, and a memory.

[0114] The memory can include a non-persistent memory, a random access memory (RAM), a non-volatile memory, and/or another form that are in a computer-readable medium, for example, a read-only memory (ROM) or a flash memory (flash RAM). The memory is an example of the computer-readable medium.

[0115] The computer-readable medium includes persistent, non-persistent, movable, and unmovable media that can store information by using any method or technology. The information can be a computer-readable instruction, a data structure, a program module, or other data. Examples of the computer storage medium include, but are not limited to, a phase change random access memory (PRAM), a static RAM (SRAM), a dynamic RAM (DRAM), a RAM of another type, a ROM, an electrically erasable programmable ROM (EEPROM), a flash memory or another memory technology, a compact disc ROM (CD-ROM), a digital versatile disc (DVD) or another optical storage, a cassette, and a cassette magnetic disk storage, or another magnetic storage device or any other non-transmission medium. The computer storage medium can be configured to store information that can be accessed by a computing device. As described in this specification, the computer-readable medium does not include transitory media such as a modulated data signal and a carrier.

[0116] It is worthwhile to further note that, the terms "include", "contain", or their any other variants are intended to cover a non-exclusive inclusion, so a process, a method, a product, or a device that includes a list of elements not only includes those elements but also includes other elements which are not expressly listed, or further includes elements inherent to such process, method, product, or device. Without more constraints, an element preceded by "includes a ..." does not preclude the existence of additional identical elements in the process, method, product, or device that includes the element.

[0117] A person skilled in the art should understand that the embodiment of this application can be provided as a method, a system, or a computer program product. Therefore, this application can use a form of hardware only embodiments, software only embodiments, or embodiments with a combination of software and hardware. Moreover, this application can use a form of a computer program product that is implemented on one or more computer-usable storage media (including but not limited to a disk memory, a CD-ROM, an optical memory, etc.) that include computer-usable program code.

[0118] This application can be described in the general context of computer-

executable instructions, for example, a program module. Generally, the program module includes a routine, a program, an object, a component, a data structure, etc. executing a specific task or implementing a specific abstract data type. This application can alternatively be practiced in distributed computing environments in which tasks are performed by remote processing devices that are connected through a communications network. In a distributed computing environment, the program module can be located in both local and remote computer storage media including storage devices.

[0119] The previous descriptions are merely embodiments of this application, and are not intended to limit this application. A person skilled in the art can make various modifications and changes to this application. Any modification, equivalent replacement, or improvement made without departing from the spirit and principle of this application shall fall within the scope of the claims in this application.

CLAIMS

What is claimed is:

1. A model gradient determining method based on federated learning, comprising:
obtaining data volume information of a participating node, wherein the data volume information is used to indicate an amount of data used by the participating node to train a basic training model based on local data, and the local data comprises user data of a target organization corresponding to the participating node;

obtaining a node local gradient obtained by training the basic training model based on the local data by the participating node;

determining, based on the data volume information and the node local gradient, a global gradient of a federated learning model that the participating node participates in;

determining a degree of participation of the participating node based on the node local gradient of the participating node and the global gradient, wherein the degree of participation is used to indicate a degree of participation of the participating node in federated learning model training; and

determining an actual model gradient of the participating node based on the degree of participation.

2. The method according to claim 1, wherein after the obtaining a node local gradient obtained by training the basic training model based on the local data by the participating node, the method further comprises:

obtaining a marginal loss of the participating node, wherein the marginal loss is used to represent a degree of influence of the node local gradient of the participating node on performance of the federated learning model; and

determining node mass of the participating node based on the marginal loss; and

the determining, based on the data volume information and the node local gradient, a global gradient of a federated learning model that the participating node participates in specifically comprises:

determining, based on the data volume information, the node local gradient, and the node mass, the global gradient of the federated learning model that the participating

node participates in.

3. The method according to claim 2, wherein the participating node participating in the federated learning model comprises a plurality of participating nodes; and

the obtaining a marginal loss of the participating node specifically comprises:

determining a first reference global model based on a node local gradient of each participating node in the plurality of participating nodes;

determining a second reference global model based on a node local gradient of each participating node other than the participating node in the plurality of participating nodes;

determining a first model loss of the first reference global model based on a predetermined verification set;

determining a second model loss of the second reference global model based on the predetermined verification set; and

determining the marginal loss of the participating node based on the first model loss and the second model loss.

4. The method according to claim 2, wherein the participating node participating in the federated learning model comprises a plurality of participating nodes; and

the determining node mass of the participating node based on the marginal loss specifically comprises:

determining the node mass of the participating node based on a marginal loss of each participating node in the plurality of participating nodes and a normalization algorithm.

5. The method according to claim 2, wherein the participating node participating in the federated learning model comprises a plurality of participating nodes;

the method further comprises:

determining a participating node with a marginal loss greater than or equal to a predetermined loss threshold in the plurality of participating nodes as an effective participating node; and

the determining, based on the data volume information, the node local gradient, and the node mass, the global gradient of the federated learning model that the

participating node participates in specifically comprises:

performing an aggregation operation on a node local gradient of each of the effective participating node based on data volume information of the participating node and node mass of the participating node, to obtain the global gradient.

6. The method according to claim 1, wherein the participating node participating in the federated learning model comprises a plurality of participating nodes; and

the determining a degree of participation of the participating node based on the node local gradient of the participating node and the global gradient specifically comprises:

determining a node contribution degree of each of the plurality of participating nodes based on a node local gradient of the participating node and the global gradient; and

determining a relative contribution degree of the participating node based on a node contribution degree of the participating node and the node contribution degree of each participating node.

7. The method according to claim 6, wherein the global gradient comprises a gradient obtained through a target quantity of rounds of iterative calculations; and

the method further comprises:

obtaining a trustworthiness parameter of the participating node, wherein the trustworthiness parameter is used to represent a comprehensive degree of reliability of the participating node in the target quantity of rounds of calculation processes of the global gradient; and

determining a reputation degree of the participating node based on the relative contribution degree and the trustworthiness parameter.

8. The method according to claim 7, wherein the obtaining a trustworthiness parameter of the participating node specifically comprises:

determining a first quantity of times that the participating node is determined as an effective participating node in the target quantity of rounds, wherein the effective participating node represents a participating node with a marginal loss greater than or equal to a predetermined loss threshold;

determining a second quantity of times that the participating node is determined as an ineffective participating node in the target quantity of rounds, wherein the ineffective participating node represents a participating node with a marginal loss less than the predetermined loss threshold; and

determining the trustworthiness parameter of the participating node based on the first quantity of times and the second quantity of times.

9. The method according to claim 7, wherein the global gradient comprises a plurality of global gradient factors, the participating node participating in the federated learning model comprises a plurality of participating nodes, and the global gradient comprises the gradient obtained through the target quantity of rounds of iterative calculations; and

the determining an actual model gradient of the participating node based on the degree of participation specifically comprises:

determining a quantity of matching gradients corresponding to the participating node based on a ratio of the reputation degree of the participating node to the greatest reputation degree, wherein the greatest reputation degree is used to represent the greatest reputation degree in reputation degrees of the plurality of participating nodes; and

selecting global gradient factors of the quantity of matching gradients from the global gradient to obtain the actual model gradient.

10. The method according to claim 9, wherein the global gradient comprises a plurality of global gradient factors, and the node local gradient comprises a plurality of local gradient factors;

the method further comprises:

obtaining a node influence degree of each global gradient factor relative to the participating node, wherein the node influence degree is used to indicate a degree that each global gradient factor is influenced by the participating node; and

sorting the global gradient factors in the global gradient based on the node influence degree, to obtain the sorted global gradient factors; and

the selecting global gradient factors of the quantity of matching gradients from the

global gradient to obtain the actual model gradient specifically comprises:

selecting global gradient factors of the quantity of matching gradients from the sorted global gradient factors based on a predetermined order, to obtain the actual model gradient.

11. The method according to claim 10, wherein the obtaining a node influence degree of each global gradient factor relative to the participating node specifically comprises:

determining a first distribution parameter corresponding to each local gradient factor in the local gradient of the participating node, wherein the first distribution parameter is used to indicate a proportion of each local gradient factor in the local gradient;

determining a second distribution parameter corresponding to each global gradient factor in the global gradient, wherein the second distribution parameter is used to indicate a proportion of each global gradient factor in the global gradient; and

determining the node influence degree of each global gradient factor relative to the participating node based on the first distribution parameter and the second distribution parameter.

12. A model gradient determining apparatus based on federated learning, comprising:

a volume information acquisition module, configured to obtain data volume information of a participating node, wherein the data volume information is used to indicate an amount of data used by the participating node to train a basic training model based on local data, and the local data comprises user data of a target organization corresponding to the participating node;

a local gradient acquisition module, configured to obtain a node local gradient obtained by training the basic training model based on the local data by the participating node;

a global gradient determining module, configured to determine, based on the data volume information and the node local gradient, a global gradient of a federated learning model that the participating node participates in;

a degree-of-participation determining module, configured to determine a degree of participation of the participating node based on the node local gradient of the participating node and the global gradient, wherein the degree of participation is used to indicate a degree of participation of the participating node in federated learning model training; and

an actual gradient determining module, configured to determine an actual model gradient of the participating node based on the degree of participation.

13. A model gradient determining device based on federated learning, comprising:
at least one processor; and

a memory communicatively connected to the at least one processor, wherein

the memory stores an instruction that can be executed by the at least one processor, and the instruction is executed by the at least one processor so that the at least one processor can:

obtain data volume information of a participating node, wherein the data volume information is used to indicate an amount of data used by the participating node to train a basic training model based on local data, and the local data comprises user data of a target organization corresponding to the participating node;

obtain a node local gradient obtained by training the basic training model based on the local data by the participating node;

determine, based on the data volume information and the node local gradient, a global gradient of a federated learning model that the participating node participates in;

determine a degree of participation of the participating node based on the node local gradient of the participating node and the global gradient, wherein the degree of participation is used to indicate a degree of participation of the participating node in federated learning model training; and

determine an actual model gradient of the participating node based on the degree of participation.

14. A computer-readable medium, storing a computer-readable instruction, wherein the computer-readable instruction can be executed by a processor to implement the model gradient determining method based on federated learning according to any

one of claims 1 to 11.