METHOD, APPARATUS, AND DEVICE FOR OBTAINING ARTIFICIAL INTELLIGENCE MODEL, AND STORAGE MEDIUM

[0001] This application claims priority to Chinese Patent Application No. 202010246686.8, filed on March 31, 2020 and entitled "METHOD, APPARATUS, AND DEVICE FOR OBTAINING ARTIFICIAL INTELLIGENCE MODEL, AND STORAGE MEDIUM", which is incorporated herein by reference in its entirety.

TECHNICAL FIELD

[0002] This application relates to the field of artificial intelligence technologies, and in particular, to a method, an apparatus, and a device for obtaining an artificial intelligence model, and a storage medium.

BACKGROUND

[0003] With development of artificial intelligence technologies, there are more artificial intelligence (artificial intelligence, AI) models, and more manners for obtaining an artificial intelligence model, such as a federated learning (federated learning, FL) manner for obtaining an artificial intelligence model.

[0004] Federated learning is an emerging basic artificial intelligence technology, is originally used to resolve a problem that a user of an Android mobile phone terminal updates a model locally, and aims to implement efficient machine learning among a plurality of participants or computing nodes on the premise of protecting terminal data and personal data privacy and ensuring legal compliance. Currently, the federated learning is expanded to jointly build an AI model without data sharing, to improve AI model effects.

SUMMARY

[0005] Embodiments of this application provide a method, an apparatus, and a device for

obtaining an artificial intelligence model, and a storage medium, to resolve a problem in a related technology. Technical solutions are as follows.

[0006] According to a first aspect, a method for obtaining an artificial intelligence model is provided. That a client performs the method is used as an example. The method includes: The client receives a first artificial intelligence AI model sent by a service end, where the first AI model includes a plurality of neurons; the client determines, from the plurality of neurons, a target neuron participating in a current round of training, where the current round of training is a non-first round of training, and a quantity of target neurons is less than a total quantity of the plurality of neurons; the client trains the target neuron based on local data; and the client returns parameter data corresponding to the target neuron to the service end, where the parameter data corresponding to the target neuron is used by the service end to obtain a converged target AI model.

[0007] According to the method provided in this embodiment of this application, not all neurons in the model need to be trained, but some neurons are selected for training based on an active condition. This can reduce power consumption, and increase a training speed. In addition, because the client transmits only a parameter corresponding to a trained neuron to the service end, communication bandwidth can be reduced.

[0008] In an example embodiment, the target neuron is a neuron whose activity degree meets a condition. For example, an activity degree of the target neuron is greater than an activity degree threshold.

[0009] In an example embodiment, each neuron in the plurality of neurons has a corresponding frozen-active flag bit, and when a value of the frozen-active flag bit is a first value, the frozen-active flag bit indicates that the neuron participates in the current round of training. The determining, from the plurality of neurons, a target neuron participating in a current round of training includes determining the neuron whose value of frozen-active flag bit is the first value as the target neuron participating in the current round of training.

[0010] A frozen-active flag bit indicates an active condition, so that a target neuron can be directly determined by using the frozen-active flag bit. Therefore, a speed of determining the target neuron is relatively high. This further reduces training time and increases the training speed.

[0011] In an example embodiment, each neuron in the plurality of neurons has a corresponding frozen period, and the frozen period indicates a period in which the neuron does not participate in training; and

a value of a frozen-active flag bit corresponding to any neuron is determined based on a frozen period corresponding to the any neuron.

[0012] In an example embodiment, before the determining the neuron whose value of frozen-active flag bit is the first value as the target neuron participating in the current round of training, the method further includes: for the any neuron in the plurality of neurons, obtaining an activity degree of the any neuron in a previous round of training, where the activity degree indicates a degree to which the neuron is affected by the local data; and

updating the frozen period of the any neuron based on the activity degree of the any neuron in the previous round of training, and updating the value of the frozen-active flag bit of the any neuron based on ~~the~~ an updated frozen period of the any neuron.

[0013] In an example embodiment, each neuron in the plurality of neurons further has a corresponding frozen period counter, and the frozen period counter indicates whether the frozen period ends; and

the obtaining an activity degree of the any neuron in a previous round of training includes: in response to the fact that a value of a frozen period counter of the any neuron indicates that the frozen period of the any neuron ends, obtaining the activity degree of the any neuron in the previous round of training.

[0014] In an example embodiment, the determining, from the plurality of neurons, a target neuron participating in a current round of training includes:

for any neuron in the plurality of neurons, obtaining an activity degree of the any neuron in a previous round of training, where the activity degree indicates a degree to which the neuron is affected by the local data; and

in response to the fact that the activity degree of the any neuron in the previous round of training is greater than an activity degree threshold, determining the any neuron as the target neuron participating in the current round of training.

[0015] In an example embodiment, the obtaining an activity degree of the any neuron in a previous round of training includes:

obtaining a first average value of parameters obtained before the previous round of training of the any neuron and a second average value of parameters obtained after the previous round of training of the any neuron;

obtaining a difference between the first average value and the second average value;

and

determining the activity degree of the any neuron in the previous round of training based on an absolute value of the difference and an absolute value of the first average value.

[0016] In an example embodiment, before the receiving a first artificial intelligence AI model sent by a service end, the method further includes:

receiving the first artificial intelligence AI model sent by the service end, determining all neurons in the plurality of neurons as target neurons, and training the target neurons based on the local data.

[0017] In an example embodiment, the returning parameter data corresponding to the target neuron to the service end further includes:

sending the frozen-active flag bit of each neuron to the ~~server~~service end, or sending, to the ~~server~~service end, a frozen-active flag bit whose value changes.

[0018] In an example embodiment, the returning parameter data corresponding to the target neuron to the service end includes: returning only the parameter data corresponding to the target neuron to the service end.

[0019] A method for obtaining an artificial intelligence model is provided. The method includes:

obtaining, by a service end, a to-be-trained first artificial intelligence AI model, where the first AI model includes a plurality of neurons;

sending, by the service end, the first AI model to a plurality of clients;

receiving, by the service end, parameter data that corresponds to a target neuron and that is returned by each client in the plurality of clients, where parameter data that corresponds to the target neuron and that is returned by any client is obtained by the any client by training the target neuron in the first AI model, and a quantity of target neurons is less than a total quantity of the plurality of neurons; and

restoring, by the service end based on the parameter data that corresponds to the target neuron and that is returned by each client, a second AI model corresponding to each client, and obtaining a converged target AI model based on the second AI model corresponding to each client.

[0020] In an example embodiment, the restoring, based on the parameter data that corresponds to the target neuron and that is returned by each client, a second AI model corresponding to each client includes:

for the any client, updating, based on the parameter data that corresponds to the target neuron and that is returned by the any client, a parameter corresponding to the target neuron in the first AI model, to obtain a second AI model corresponding to the any client.

[0021] In an example embodiment, the obtaining a converged target AI model based on the second AI model corresponding to each client includes:

performing federated averaging on the second AI model corresponding to each client, to obtain a third AI model; and

in response to the fact that the third AI model is converged, using the third AI model as the target AI model; or

in response to the fact that the third AI model is not converged, sending the third AI model to the plurality of clients, continuing to obtain a new AI model in a manner of obtaining the third AI model, and repeating this process until the converged target AI model is obtained.

[0022] In an example embodiment, before the restoring, based on the parameter data that corresponds to the target neuron and that is returned by each client, a second AI model corresponding to each client, the method further includes:

receiving a frozen-active flag bit of each neuron or a frozen-active flag bit whose value changes, that is returned by the plurality of clients; and

determining the target neuron in the plurality of neurons by using the frozen-active flag bit of each neuron or the frozen-active flag bit whose value changes.

[0023] An apparatus for obtaining an artificial intelligence model is provided. The apparatus includes:

a communications unit, configured to receive a first artificial intelligence AI model sent by the service end, where the first AI model includes a plurality of neurons; and

a processing unit, configured to determine, from the plurality of neurons, a target neuron participating in a current round of training, where the current round of training is a non-first round of training, and a quantity of target neurons is less than a total quantity of the plurality of neurons.

[0024] The processing unit is further configured to train the target neuron based on local data.

[0025] The communications unit is further configured to return parameter data corresponding to the target neuron to the service end. The parameter data corresponding to the target neuron is used by the service end to obtain a converged target AI model.

[0026] In an example embodiment, each neuron in the plurality of neurons has a corresponding frozen-active flag bit, and when a value of the frozen-active flag bit is a first value, the frozen-active flag bit indicates that the neuron participates in the current round of training.

[0027] The processing unit is configured to determine the neuron whose value of frozen-active flag bit is the first value as the target neuron participating in the current round of training.

[0028] In an example embodiment, each neuron in the plurality of neurons further has a corresponding frozen period, and the frozen period indicates a period in which the neuron does not participate in training.

[0029] A value of a frozen-active flag bit corresponding to any neuron is determined based on a frozen period corresponding to the any neuron.

[0030] In an example embodiment, the processing unit is further configured to: for the any neuron in the plurality of neurons, obtain an activity degree of the any neuron in a previous round of training, where the activity degree indicates a degree to which the neuron is affected by the local data; and update the frozen period of the any neuron based on the activity degree of the any neuron in the previous round of training, and update the value of the frozen-active flag bit of the any neuron based on ~~the~~ an updated frozen period of the any neuron.

[0031] In an example embodiment, each neuron in the plurality of neurons further has a corresponding frozen period counter, and the frozen period counter indicates whether the frozen period ends.

[0032] The processing unit is configured to: in response to the fact that a value of a frozen period counter of the any neuron indicates that the frozen period of the any neuron ends, obtain the activity degree of the any neuron in the previous round of training.

[0033] In an example embodiment, the processing unit is configured to: in response to the fact that the current round of training is the non-first round of training, for any neuron in the plurality of neurons, obtain an activity degree of the any neuron in a previous round of training, where the activity degree indicates a degree to which the neuron is affected by the local data; and in response to the fact that the activity degree of the any neuron in the previous round of training is greater than an activity degree threshold, determine the any neuron as the target neuron participating in the current round of training.

[0034] In an example embodiment, the processing unit is configured to: obtain a first average value of parameters obtained before the previous round of training of the any neuron and a second

average value of parameters obtained after the previous round of training of the any neuron; obtain a difference between the first average value and the second average value; and determine the activity degree of the any neuron in the previous round of training based on an absolute value of the difference and an absolute value of the first average value.

[0035] In an example embodiment, the processing unit is further configured to: receive the first artificial intelligence AI model sent by the service end, determine all neurons in the plurality of neurons as target neurons, and train the target neurons based on the local data.

[0036] In an example embodiment, the communications unit is further configured to send the frozen-active flag bit of each neuron to the service end~~server~~, or send, to the service end~~server~~, a frozen-active flag bit whose value changes.

[0037] In an example embodiment, the communications unit is configured to return only the parameter data corresponding to the target neuron to the service end.

[0038] An apparatus for obtaining an artificial intelligence model is further provided. The apparatus includes:

a processing unit, configured to obtain a to-be-trained first artificial intelligence AI model, where the first AI model includes a plurality of neurons; and

a communications unit, configured to send the first AI model to a plurality of clients.

[0039] The communications unit is further configured to receive parameter data that corresponds to a target neuron and that is returned by each client in the plurality of clients. Parameter data that corresponds to the target neuron and that is returned by any client is obtained by the any client by training the target neuron in the first AI model, and a quantity of target neurons is less than a total quantity of the plurality of neurons.

[0040] The processing unit is further configured to: restore, based on the parameter data that corresponds to the target neuron and that is returned by each client, a second AI model corresponding to each client, and obtain a converged target AI model based on the second AI model corresponding to each client.

[0041] In an example embodiment, the processing unit is configured to: for any client, update, based on the parameter data that corresponds to the target neuron and that is returned by the any client, a parameter corresponding to the target neuron in the first AI model, and supplement a parameter corresponding to a non-target neuron in the first AI model, to obtain a second AI model corresponding to the any client.

[0042] In an example embodiment, the processing unit is configured to: perform federated averaging on the second AI model corresponding to each client, to obtain a third AI model; and in response to the fact that the third AI model is converged, use the third AI model as the target AI model; or in response to the fact that the third AI model is not converged, send the third AI model to the plurality of clients, continue to obtain a new AI model in a manner of obtaining the third AI model, and repeat this process until the converged target AI model is obtained.

[0043] In an example embodiment, the communications unit is further configured to receive a frozen-active flag bit of each neuron or a frozen-active flag bit whose value changes, that is returned by the plurality of clients.

[0044] The processing unit is configured to determine the target neuron in the plurality of neurons by using the frozen-active flag bit of each neuron or the frozen-active flag bit whose value changes.

[0045] A device for obtaining an artificial intelligence model is further provided. The device includes a memory and a processor. The memory stores at least one instruction, and the at least one instruction is loaded and executed by the processor, to implement the method for obtaining an artificial intelligence model according to any one of the first aspect and the example embodiments of the first aspect.

[0046] A device for obtaining an artificial intelligence model is further provided. The device includes a memory and a processor. The memory stores at least one instruction, and the at least one instruction is loaded and executed by the processor, to implement the method for obtaining an artificial intelligence model according to any one of the second aspect and the example embodiments of the second aspect.

[0047] A computer-readable storage medium is further provided. The storage medium stores at least one instruction, and the instruction is loaded and executed by a processor to implement any one of the foregoing methods for obtaining an artificial intelligence model.

[0048] Another communications apparatus is provided. The apparatus includes a transceiver, a memory, and a processor. The transceiver, the memory, and the processor communicate with each other through an internal connection channel. The memory is configured to store instructions. The processor is configured to execute the instructions stored in the memory, to control the transceiver to receive a signal, and control the transceiver to send a signal. In addition, when the processor executes the instructions stored in the memory, the processor is enabled to perform the

method according to the first aspect and the possible implementations of the first aspect.

[0049] Another communications apparatus is provided. The apparatus includes a transceiver, a memory, and a processor. The transceiver, the memory, and the processor communicate with each other through an internal connection channel. The memory is configured to store instructions. The processor is configured to execute the instructions stored in the memory, to control the transceiver to receive a signal, and control the transceiver to send a signal. In addition, when the processor executes the instructions stored in the memory, the processor is enabled to perform the method according to the second aspect and the possible implementations of the second aspect.

[0050] In an example embodiment, there are one or more processors, and there are one or more memories.

[0051] In an example embodiment, the memory may be integrated with the processor, or the memory is disposed independently of the processor.

[0052] In a specific implementation process, the memory may be a non-transitory (non-transitory) memory, such as a read-only memory (read-only memory, ROM). The memory and the processor may be integrated into one chip, or may be separately disposed in different chips. A type of the memory and a manner in which the memory and the processor are disposed are not limited in this embodiment of this application.

[0053] A computer program (product) is provided. The computer program (product) includes computer program code. When the computer program code is run on a computer, the computer is enabled to perform the methods according to the foregoing aspects.

[0054] A chip is provided. The chip includes a processor, configured to: invoke, from a memory, instructions stored in the memory and run the instructions, so that a communications device on which the chip is installed performs the methods according to the foregoing aspects.

[0055] Another chip is provided. The chip includes an input interface, an output interface, a processor, and a memory. The input interface, the output interface, the processor, and the memory are connected to each other through an internal connection channel. The processor is configured to execute code in the memory. When the code is executed, the processor is configured to perform the methods according to the foregoing aspects.

BRIEF DESCRIPTION OF DRAWINGS

[0056] FIG. 1 is a schematic diagram of a network architecture for federated learning according to an embodiment of this application;

[0057] FIG. 2 is a schematic diagram of a network architecture for federated learning according to an embodiment of this application;

[0058] FIG. 3 is a flowchart of a method for obtaining an artificial intelligence model according to an embodiment of this application;

[0059] FIG. 4 is a schematic diagram of an interaction process of obtaining an artificial intelligence model according to an embodiment of this application;

[0060] FIG. 5 is a schematic diagram of a process of obtaining an artificial intelligence model on a client according to an embodiment of this application;

[0061] FIG. 6 is a schematic diagram of a structure of an apparatus for obtaining an artificial intelligence model according to an embodiment of this application;

[0062] FIG. 7 is a schematic diagram of a structure of an apparatus for obtaining an artificial intelligence model according to an embodiment of this application; and

[0063] FIG. 8 is a schematic diagram of a structure of a device for obtaining an artificial intelligence model according to an embodiment of this application.

DESCRIPTION OF EMBODIMENTS

[0064] Terms used in an implementation part of this application are merely used to explain embodiments of this application, and are not intended to limit this application.

[0065] With the development of artificial intelligence technologies, there are more and more artificial intelligence models, and manners of obtaining the artificial intelligence models are also increasing, such as a federated learning manner for obtaining an AI model. Federated learning is an emerging basic artificial intelligence technology, is originally used to resolve a problem that a user of an Android mobile phone terminal updates a model locally, and aims to implement efficient machine learning among a plurality of participants or computing nodes on the premise of protecting terminal data and personal data privacy and ensuring legal compliance. Currently, the federated learning is expanded to jointly build an AI model without data sharing, to improve AI model effects.

[0066] A network architecture for federated learning shown in FIG. 1 is used as an example. The network architecture is divided into two layers: A lower layer is a plurality of clients, and an upper layer is a service end. The client usually indicates a device, and a large amount of data flows into the device. As shown in FIG. 1, an example in which the plurality of clients included in the network architecture are a client 1, a client 2, and a client 3 is used. Usually, it is difficult to collect or share data of the clients. Therefore, an AI model trained based on data of the client 1 is valid only for the client 1, an AI model trained based on data of the client 2 is valid only for the client 2, and an AI model trained based on data of the client 3 is valid only for the client 3. To use data from all clients, a federated learning manner for obtaining an AI model emerges.

[0067] In a process of obtaining an AI model through federated learning, each client starts local training on an initial central model (generally the same) by using data of the current device (data of each client may be different). After local training is performed on each client for a plurality of batches, local models are uploaded to the service end. The service end performs aggregation and federated update on the local models to form an updated central model. If the updated central model is not converged, the service end continues to deliver the updated central model to each client. Each client continues local training on the updated central model by using the data of the current device, and then uploads the updated central model to the service end. This process repeats until a converged central model is obtained. In this case, federated training is completed.

[0068] The central model obtained through training in the foregoing federated learning manner has an excellent discrimination capability for data of all clients, which greatly meets a requirement that an AI model can be jointly built by collecting data of different clients.

[0069] However, in the foregoing process of training an AI model through federated learning, each client needs to perform complete training on a complete model delivered by the service end, and training time is relatively long. In addition, the complete model is transmitted between the client and the service end, resulting in relatively high communication overheads.

[0070] Therefore, an embodiment of this application provides a method for obtaining an AI model. In the method, when local training is performed on a client, only neurons that can be active by being affected by local data are trained, and inactive neurons are frozen. That inactive neurons are frozen indicates that the neurons do not participate in training. Because the client uploads only parameter data of an active target neuron each time, to transmit important information to the service end, even for a small-scale client, the service end does not lose converged important

information. In addition, because only the non-frozen target neuron is trained during training, training time of the client is accelerated, and power consumption of the device is reduced.

[0071] For example, the method provided in this embodiment of this application may be applied to a network architecture shown in FIG. 2. As shown in FIG. 2, the network architecture includes two clients: a client 1 and a client 2. The client 1 and the client 2 may be any device that supports local training, including but not limited to: a mobile phone, a tablet, a computer, a switch, an optical line terminal (optical line terminal, OLT), an optical network terminal (optical network terminal, ONT), a router, or the like. A service end may be any device or cloud platform that supports multi-model aggregation and federation, including but not limited to a server, a public cloud, or the like.

[0072] As shown in FIG. 2, each client performs a current round of model training by using local data, trains only a target neuron that participates in the current round of training, and uploads corresponding parameter data of the target neuron to the service end after the training ends. For example, to the service end, a local model is uploaded, or only a parameter corresponding to the target neuron that participates in the current round of training is uploaded, or a parameter corresponding to each neuron (including a parameter corresponding to the target neuron and a parameter corresponding to a non-target neuron) is uploaded, or only a parameter difference corresponding to the target neuron is uploaded. After aggregating parameter data that corresponds to the target neuron participating in training and that is returned by each client, the service end restores an AI model of each client, and obtains an updated central model based on the AI model. If the updated central model is not converged, the updated central model is delivered to each client. Each client starts a next round of training until a converged central model is obtained.

[0073] The following uses the network architecture shown in FIG. 2 as an example to describe the method for obtaining an artificial intelligence model provided in this embodiment of this application. As shown in FIG. 3, in the method, an AI model may be obtained through interaction between a service end and a client. The method provided in this embodiment of this application may include the following several processes.

[0074] 301: The service end obtains a to-be-trained first AI model, where the first AI model includes a plurality of neurons.

[0075] The to-be-trained first AI model may be an initial AI model that is not trained, or may be an AI model that has undergone one or more rounds of training but has not converged and still

needs to be retrained. Regardless of the initial AI model or the AI model that has not converged and needs to be retrained, the first AI model includes a plurality of neurons. A neuron is a basic unit of an AI model, mainly simulates a structure and characteristics of a biological neuron, receives a group of input signals, and generates output.

[0076] A quantity of neurons included in the first AI model is not limited in this embodiment of this application, and a type of the to-be-trained first AI model is not limited in this embodiment of this application. The first AI model may be any type of AI model. For example, when obtaining the to-be-trained first AI model, a server may determine, based on an application scenario, a type of the first AI model to be obtained.

[0077] **302: The service end sends the first AI model to a plurality of clients.**

[0078] In an implementation environment to which the method provided in this embodiment of this application is applied, as shown in FIG. 2, the service end is communicatively connected to the plurality of clients, and the plurality of clients may perform federated learning to obtain a final converged target AI model. Therefore, the service end obtains the first AI model, the service end may send the first AI model to the plurality of clients, and the plurality of clients separately perform a training process on the first AI model.

[0079] In an example embodiment, when sending the first AI model to the client, the service end may further send a quantity of training batches to the client, so that each client can determine a quantity of times of performing a current round of training on the first AI model. If a plurality of rounds of training are required, the service end may send a corresponding quantity of batches each time the service end sends the first AI model to the client, and quantities of batches for different rounds of training may be different or the same. If the quantities of batches for the rounds of training are the same, the service end may deliver the quantity of batches to the client in only a first round of training, and does not repeatedly send the quantity of batches subsequently. Furthermore, in addition to a manner in which the service end sends the quantity of training batches to the client, the service end may not deliver the quantity of training batches, but each round of training is performed based on a quantity of batches that is agreed on in advance.

[0080] **303: The client receives the first AI model sent by the service end.**

[0081] The client may be any one of the plurality of clients connected to the service end.

[0082] It should be noted that, in this embodiment of this application, only one of the plurality of clients connected to the service end is used as an example to describe an operation process of

the client. The operation process of the client in this embodiment of this application is applicable to the plurality of clients connected to the service end, and operation processes of all the clients are not described one by one again.

[0083] 304: The client determines, from the plurality of neurons, a target neuron participating in the current round of training. A quantity of target neurons is less than a total quantity of the plurality of neurons, and the current round of training is a non-first round of training.

[0084] Because local data has different degrees of impact on different neurons, parameters of some neurons may not change greatly after local data–based training. Therefore, in the method provided in this embodiment of this application, after receiving the to-be-trained first AI model, the client may not need to train all neurons in the first AI model, but train only a target neuron whose activity degree meets a specific condition. The target neuron whose activity degree meets the specific condition may be a neuron that can be active due to impact of the local data. A manner in which the client determines, from the plurality of neurons, the target neuron participating in the current round of training is not limited in this embodiment of this application. For example, the target neuron is a neuron whose activity degree meets a condition. For example, the activity degree is greater than an activity degree threshold.

[0085] For the first round of training, because the first AI model is not trained, which neurons are active and which neurons are inactive cannot be determined. Therefore, the determining, from the plurality of neurons, a target neuron participating in the current round of training includes: receiving the first artificial intelligence AI model sent by the service end, determining all of the plurality of neurons as target neurons, and training the target neurons based on the local data.

[0086] For a case in which the current round of training is the non-first round of training, only the following two manners are used as examples for description.

[0087] Manner 1: The target neuron participating in the current round of training is determined in the plurality of neurons based on an activity degree of the neuron.

[0088] For the case in which the current round of training is the non-first round of training, because the neurons are trained in a previous round of training by using the local data, which neurons are active and which neurons are inactive may be determined. Therefore, for the case in which the current round of training is the non-first round of training, the determining, from the plurality of neurons, a target neuron participating in the current round of training includes but is

not limited to: for any neuron in the plurality of neurons, obtaining an activity degree of the any neuron in the previous round of training, where the activity degree indicates a degree to which the neuron is affected by the local data; and in response to the fact that the activity degree of the any neuron in the previous round of training is greater than the activity degree threshold, using the any neuron as the target neuron participating in the current round of training.

[0089] A manner of obtaining the activity degree of the any neuron in the previous round of training is not limited in this embodiment of this application. For example, the obtaining an activity degree of the any neuron in the previous round of training includes: obtaining a first average value of parameters obtained before the previous round of training of the any neuron and a second average value of parameters obtained after the previous round of training of the any neuron; obtaining a difference between the first average value and the second average value; and determining, based on an absolute value of the difference and an absolute value of the first average value, the activity degree of the any neuron in the previous round of training.

[0090] For example, the first average value of the parameters obtained before the previous round of training of the any neuron is AC, and the second average value of the parameters obtained after the previous round of training of the any neuron is AE. A quotient of the absolute value of the difference and the absolute value of the first average value is determined as the activity degree of the any neuron in the previous round of training. For example, the activity degree AR of the any neuron in the previous round of training is obtained by using the following formula:

$$AR = \frac{|AE-AC|}{|AC|}, \text{ or } AR = \frac{|AE-AC|}{|AC|+x}$$

[0091] In addition to the foregoing manner of obtaining the activity degree of the any neuron in the previous round of training, a variation may be further performed in the foregoing manner, to obtain another optional manner of obtaining the activity degree. x is a minimum value that prevents a quotient from being zero, for example, may be 0.0001. A value of x is not limited in this embodiment of this application.

[0092] For example, the first average value of the parameters obtained before the previous round of training of the any neuron is AC, and the second average value of the parameters obtained after the previous round of training of the any neuron is AE. The activity degree AR of the any neuron in the previous round of training is obtained by using the following formula:

$$AR = \frac{|AE-AC|}{|AC|} * \mu, \text{ or } AR = \frac{|AE-AC|}{|AC|+x} * \mu$$

[0093] In the formula, μ is an activity degree coefficient, and may be set based on experience or may be adjusted based on a status of each round of training.

[0094] For another example, the first average value of the parameters obtained before the previous round of training of the any neuron is AC, and the second average value of the parameters obtained after the previous round of training of the any neuron is AE. The activity degree AR of the any neuron in the previous round of training is obtained by using the following formula:

$$AR = \frac{|AE-AC|}{|AC|} + \varphi, \text{ or } AR = \frac{|AE-AC|}{|AC|+x} + \varphi$$

[0095] In the formula, φ is an activity degree constant, and may be set based on experience or may be adjusted based on a status of each round of training.

[0096] Regardless of which manner is used to obtain the activity degree of the any neuron, the obtained activity degree may be compared with the activity degree threshold. If the activity degree is greater than the activity degree threshold, the any neuron is used as the target neuron participating in the current round of training. The activity degree threshold is not limited in this embodiment of this application, and may be set based on experience or may be adjusted based on a status of each round of training.

[0097] **Manner 2: The target neuron participating in the current round of training is determined in the plurality of neurons by querying a frozen-active flag bit.**

[0098] A difference between Manner 2 and Manner 1 in which an activity degree of a neuron is calculated during each round of training, and a target neuron is determined based on whether the activity degree meets a threshold lies in that in the method provided in this embodiment of this application, the activity degree may not be calculated during the current round of training, but is implemented by querying the frozen-active flag bit. In an example embodiment, each neuron of the plurality of neurons has a corresponding frozen-active flag bit, and when a value of the frozen-active flag bit is a first value, the frozen-active flag bit indicates that the neuron participates in the current round of training. For example, after each round of training ends, the activity degree may be calculated in any one of the foregoing manners, and a value of a frozen-active flag bit of a neuron whose activity degree meets the threshold is set to the first value. In this case, the determining, from the plurality of neurons, a target neuron participating in the current round of training includes: determining a neuron whose value of frozen-active flag bit is the first value as the target neuron participating in the current round of training.

[0099] In the foregoing manner of indicating, by using the frozen-active flag bit, whether to

participate in the current round of training, although the value of the frozen-active flag bit also needs to be determined by using the activity degree, because the value is obtained through calculation after the previous round of training ends, namely, obtained before the current round of training, the target neuron can be directly determined by using the frozen-active flag bit during the current round of training, so that a speed of determining the target neuron participating in the current round of training is relatively high. This further reduces training time and increases a training speed.

[00100] Further, how the value of the frozen-active flag bit of the neuron is determined during the current round of training is not limited in this embodiment of this application. In an example embodiment, each neuron of the plurality of neurons further has a corresponding frozen period, and the frozen period indicates a period in which the neuron does not participate in training. A value of a frozen-active flag bit corresponding to any neuron is determined based on a frozen period corresponding to the any neuron. For example, if a frozen period of any neuron ends, a frozen-active flag bit of the any neuron is the first value. In this way, a detection mechanism is provided by setting a frozen period, to indicate, based on the frozen period, the period in which the neuron does not participate in training, determine, based on the frozen period, whether the neuron participates in training, and determine a value of a frozen-active flag bit, thereby preventing loss of a parameter corresponding to a neuron that needs to participate in training in a training process, and ensuring robustness.

[00101] In this embodiment of this application, before the determining the neuron whose value of frozen-active flag bit is the first value as the target neuron participating in the current round of training, the method further includes: for any one of the plurality of neurons, obtaining the activity degree of the any neuron in the previous round of training, where the activity degree indicates a degree to which the neuron is affected by the local data; and updating the frozen period of the any neuron based on the activity degree of the any neuron in the previous round of training, and updating the value of the frozen-active flag bit of the any neuron based on ~~the~~ an updated frozen period of the any neuron.

[00102] For the manner of obtaining the activity degree of the any neuron in the previous round of training, refer to the description of the activity degree calculation manner in Manner 1. Details are not described herein again. For example, the updating the frozen period of the any neuron based on the activity degree of the any neuron in the previous round of training includes but is not

limited to updating the frozen period by using the following formula:

$$\mathrm{Ti}' = \begin{cases} \mathrm{Ti}+1 & \mathrm{ARi} < X1 \\ \mathrm{Ti} & \mathrm{X1} \le \mathrm{ARi} \le \mathrm{X2} \\ \lfloor (\mathrm{Ti}+1)/2 \rfloor & \mathrm{ARi} \ge \mathrm{X2} \end{cases}$$

[00103] In the foregoing formula for updating the frozen period, Ti' indicates a frozen period after an i^{th} neuron is updated, Ti indicates a frozen period before the i^{th} neuron is updated, ARi indicates an activity degree of the i^{th} neuron in a previous round of training, i is an integer ranging from 1 to N, and N is a total quantity of neurons. X1 indicates a first activity degree threshold, X2 indicates a second activity degree threshold, and X2 is greater than X1. $\lfloor (\mathrm{Ti}+1)/2 \rfloor$ indicates that $(\mathrm{Ti}+1)/2$ is rounded down.

[00104] Sizes of the first activity degree threshold and the second activity degree threshold are not limited in this embodiment of this application, and may be set based on experience or may be adjusted based on a training status. For example, if X1 = 0.2 and X2 = 0.4, the foregoing formula for updating the frozen period is:

$$T\mathrm{i}' = \begin{cases} T\mathrm{i}+1 & AR\mathrm{i} < 0.2 \\ T\mathrm{i} & 0.2 \le AR\mathrm{i} \le 0.4 \\ \lfloor (\mathrm{Ti}+1)/2 \rfloor & AR\mathrm{i} \ge 0.4 \end{cases}$$

[00105] For example, if an activity degree of any neuron is 0.3, a value of the frozen period Ti' after the update of the neuron is the same as a value of the frozen period Ti before the update. It may be considered that the activity degree of the neuron does not change greatly, and therefore, an original frozen period may be maintained. For example, if an active degree of any neuron is 0.1, the frozen period Ti' after the update of the neuron is equal to Ti + 1. It may be considered that the neuron is not active enough, and therefore, the frozen period is increased. For example, if an active degree of any neuron is 0.6, the frozen period Ti' after the update of the neuron is equal to $\lfloor (\mathrm{Ti}+1)/2 \rfloor$. It may be considered that the neuron is relatively active, and therefore, the frozen period is reduced, so that the neuron participates in training as soon as possible, to ensure that no important information is lost.

[00106] In an example embodiment, the updating the value of the frozen-active flag bit of the any neuron based on ~~the~~ an updated frozen period of the any neuron includes but is not limited to: for a neuron whose Ti' > Y, setting Fi' to a second value, for example, setting the second value to 1, which indicates that the current round of training can be frozen, that is, the neuron does not participate in the current round of training; and for a neuron whose Ti' = Y, setting Fi' to a first value, for example, setting the first value to 0, which indicates that the current round of training is

not frozen, that is, the neuron participates in the current round of training. For example, Y may be 1.

[00107] In an example embodiment, after a round of training ends, the foregoing update process may not be performed on a frozen period of each neuron, but only a neuron whose frozen period ends is updated. Each of the plurality of neurons further has a corresponding frozen period counter, and the frozen period counter indicates whether the frozen period ends. The obtaining the activity degree of the any neuron in the previous round of training includes: in response to the fact that a value of a frozen period counter of the any neuron indicates that a frozen period of the any neuron ends, obtaining the activity degree of the any neuron in the previous round of training.

[00108] In this implementation, the frozen period counter is set, and indicates whether the frozen period ends, and the activity degree is obtained after the frozen period ends. This further reduces time for detecting whether the frozen period ends, and increases a training speed. For example, a lower activity degree indicates a larger value of a frozen period and a longer frozen period. An initial value of the frozen period counter may be set to 1. Each time a quantity of training rounds increases by one, a value of the frozen period counter increases by 1. In other words, the value of the frozen period is positively correlated with a value of the frozen period counter. When the value of the frozen period counter is equal to the value of the frozen period, it may be considered that the frozen period ends. In addition, the value of the frozen period counter may alternatively be negatively correlated with the value of the frozen period. For example, a lower activity degree indicates a larger value of a frozen period and a longer frozen period. An initial value of the frozen period counter may be set to the value of the frozen period. Each time a quantity of training rounds increases by one, the value of the frozen period counter decreases by 1. When the value of the frozen period counter is 0, it may be considered that the frozen period ends.

[00109] It should be noted that, when the current round of training is the first round of training, because an activity degree of a neuron in a previous round of training cannot be determined based on a result of the previous round of training, a value of a frozen-active flag bit of the neuron in the current round of training cannot be determined based on the activity degree of the neuron in the previous round of training. Therefore, this embodiment of this application provides an initialization process during the first round of training. For example, in response to the fact that the current round of training is the first round of training, a value of a frozen-active flag bit of each neuron in the plurality of neurons is set to the first value.

[00110] In other words, when the current round of training is the first round of training, a value of a frozen-active flag bit of each neuron is set to the first value. In this case, each neuron may be determined as a target neuron, to participate in the current round of training. Then, the value of the frozen-active flag bit of the neuron may be updated based on a result of the current round of training, so that during next round of training, the target neuron can be directly determined based on the value of the frozen-active flag bit that is updated in the current round of training. For a manner of updating the value of the frozen-active flag bit of the neuron, refer to the foregoing related descriptions. Details are not described herein again.

[00111] 305: The client trains the target neuron based on the local data.

[00112] In the method provided in this embodiment of this application, the client may train the target neuron based on the local data, which may be performed based on a batch of the current round of training. For example, if the current round of training includes E batches, after performing one batch of the current round of training on the target neuron, the client obtains an updated AI model, and then continues to perform training until the E batches of the current round of training are completed.

[00113] A quantity of batches of the current round of training may be sent by the service end, or may be agreed in advance. A manner of obtaining the quantity of batches of the current round of training and the quantity of batches are not limited in this embodiment of this application.

[00114] 306: The client returns parameter data corresponding to the target neuron to the service end.

[00115] A manner in which the client returns the parameter data corresponding to the target neuron to the service end is not limited in this embodiment of this application. Only a parameter corresponding to the target neuron may be returned, or only a parameter difference corresponding to the target neuron may be returned, thereby reducing communication bandwidth between the client and the service end. In addition, because the AI model obtained after training includes the parameter corresponding to the target neuron, the method provided in this embodiment of this application also supports returning the AI model obtained after training, to return the parameter data corresponding to the target neuron in a manner of returning the AI model obtained after training. Alternatively, parameters or parameter differences corresponding to all neurons may be returned to the service end. A parameter difference corresponding to any neuron may be a difference between a parameter that corresponds to the any neuron and that is obtained in the

current round of training and a parameter that corresponds to the any neuron and that is obtained in the previous round of training.

[00116] For a manner of indicating, by using a frozen-active flag bit, whether a neuron participates in the current round of training, in an example embodiment, that the client returns the parameter data corresponding to the target neuron to the service end further includes: sending a frozen-active flag bit of each neuron to the service end~~server~~, or sending, to the service end~~server~~, a frozen-active flag bit whose value changes. A manner of sending the frozen-active flag bit is not limited in this embodiment of this application. If the current round of training is the first round of training, the client may send the frozen-active flag bit of each neuron to the service end~~server~~. For a neuron whose value of the frozen-active flag bit does not change subsequently, the client may no longer send the frozen-active flag bit of the neuron subsequently, which further reduces communication bandwidth. For example, the manner of sending the frozen-active flag bit is not limited in this embodiment of this application. For example, if there are five neurons, five bits of data may be used for indication, and one bit corresponds to a frozen-active flag bit of one neuron. For the target neuron, a first value of the frozen-active flag bit is 1, and the frozen-active flag bit returned by the client may be 10011. If values of the first bit, the fourth bit, and the fifth bit are 1, it indicates that the first neuron, the fourth neuron, and the fifth neuron are target neurons.

[00117] 307: The service end receives the parameter data that corresponds to the target neuron and that is returned by each of the plurality of clients.

[00118] Parameter data that corresponds to the target neuron and that is returned by any client is obtained by the any client by training the target neuron in the first AI model, and a quantity of target neurons is less than a total quantity of the plurality of neurons.

[00119] It should be noted that although the service end may send a same first AI model to each of the plurality of clients, because the clients have different local data, and the different local data has different degrees of impact on the neurons, there may be a case in which target neurons that participate in the current round of training and that are determined by different clients may be inconsistent, and parameters obtained after the target neurons are trained by using the local data are also different. In other words, the parameter data that corresponds to the target neurons and that is received by the service end from the plurality of clients may be different.

[00120] For example, the network architecture shown in FIG. 2 is used as an example. If the first AI model includes four neurons: a neuron 1, a neuron 2, a neuron 3, and a neuron 4, after the

client 1 receives the first AI model sent by the service end, and determines activity degrees of the four neurons based on local data of the client 1, the client 1 determines that the activity degree of the neuron 1 and the activity degree of the neuron 2 are greater than an activity degree threshold. In other words, target neurons determined by the client 1 are the neuron 1 and the neuron 2. In this case, parameter data returned to the service end is also parameter data of the neuron 1 and parameter data of the neuron 2. After the client 2 receives the first AI model sent by the service end, and determines activity degrees of the four neurons based on local data of the client 2, the client 2 determines that the activity degree of the neuron 1, the activity degree of the neuron 2, and the activity degree of the neuron 3 are greater than the activity degree threshold. In other words, target neurons determined by the client 2 are the neuron 1, the neuron 2, and the neuron 3. In this case, parameter data returned to the service end is also parameter data of the neuron 1, parameter data of the neuron 2, and parameter data of the neuron 3.

[00121] It can be learned that a quantity of the target neurons determined by the client 1 is different from a quantity of the target neurons determined by the client 2. In addition, even if both the client 1 and the client 2 determine the neuron 1 and the neuron 2 as the target neurons, because the local data of the client 1 and the local data of the client 2 are different, for the same target neuron 1 and the neuron 2, the parameter data obtained by the client 1 by training the two neurons may also be different from the parameter data obtained by the client 2 by training the two neurons.

[00122] 308: The service end restores, based on the parameter data that corresponds to the target neuron and that is returned by each client, a second AI model corresponding to each client, and obtains a converged target AI model based on the second AI model corresponding to each client.

[00123] As described above, because the local data of the clients may be different, the parameter data that corresponds to the target neurons and that is returned by the clients to the service end may also be different. Therefore, the service end restores, based on the parameter data that corresponds to the target neuron and that is returned by each client, the second AI model corresponding to each client.

[00124] In an example embodiment, that the service end restores, based on the parameter data that corresponds to the target neuron and that is returned by each client, the second AI model corresponding to each client includes: for any client, updating, based on parameter data that corresponds to a target neuron and that is returned by the any client, a parameter corresponding to

the target neuron in the first AI model, to obtain a second AI model corresponding to the any client. For example, a non-target neuron in the second AI model is a parameter corresponding to a non-target neuron in the first AI model.

[00125] For example, the first AI model includes a neuron 1 and a neuron 2. The client 2 returns a parameter of the neuron 1, and does not return a parameter of the neuron 2. In this case, the neuron 1 is the target neuron, and the neuron 2 is the non-target neuron. A parameter corresponding to the neuron 1 in the first AI model is updated based on the parameter that corresponds to the neuron 1 and that is returned by the client, and a parameter corresponding to the neuron 2 in the first AI model is supplemented, to obtain a second AI model corresponding to the client 2.

[00126] For another example, the first AI model includes a neuron 1 and a neuron 2. The client 1 returns a parameter corresponding to the neuron 2, and does not return a parameter corresponding to the neuron 1. In this case, the neuron 2 is the target neuron, and the neuron 1 is the non-target neuron. A parameter corresponding to the neuron 2 in the first AI model is updated based on the parameter that corresponds to the neuron 2 and that is returned by the client, and a parameter corresponding to the neuron 1 in the first AI model is supplemented, to obtain a second AI model corresponding to the client 1.

[00127] A manner of supplementing a parameter corresponding to a non-target neuron is not limited in this embodiment of this application. For example, a parameter corresponding to a same non-target neuron in an AI model updated in a previous round of training may be used to replace a parameter corresponding to the non-target neuron in the first AI model.

[00128] It should be noted that, if the client returns the parameter difference corresponding to the target neuron, the value of the parameter corresponding to the target neuron in the first AI model and the parameter difference may be integrated to obtain the parameter corresponding to the target neuron.

[00129] When the second AI model corresponding to each client is obtained through restoration, to distinguish which neurons are target neurons, before the restoring the second AI model corresponding to each client based on the parameter that corresponds to the target neuron and that is returned by each client, the method provided in this embodiment of this application further includes: receiving a frozen-active flag bit of each neuron or a frozen-active flag bit whose value changes, that is returned by the plurality of clients; and determining the target neuron in the plurality of neurons by using the frozen-active flag bit of each neuron or the frozen-active flag bit

whose value changes.

[00130] In an example embodiment, the obtaining the converged target AI model based on the second AI model corresponding to each client includes: performing federated averaging on the second AI model corresponding to each client, to obtain a third AI model; and in response to the fact that the third AI model is converged, using the third AI model as the target AI model; or in response to the fact that the third AI model is not converged, sending the third AI model to the plurality of clients, continuing to obtain a new AI model in a manner of obtaining the third AI model, and repeating this process until a converged target AI model is obtained.

[00131] For example, a manner of performing federated averaging on the second AI model corresponding to each client to obtain the third AI model is not limited in this embodiment of this application. Because one neuron may have one or more parameters, a same parameter corresponding to a same neuron in each second AI model may be averaged, and the average value is used as a parameter value of the same parameter corresponding to the same neuron in the third AI model. For example, for a same neuron in the second AI model corresponding to each client, an average value of parameters corresponding to the same neuron may be used as a target parameter value, and a parameter value corresponding to the neuron in the first AI model is replaced with the target parameter value, to obtain the third AI model. For example, the first AI model includes a neuron 1 and a neuron 2. A second AI model corresponding to any client is obtained through restoration based on the first AI model. Therefore, the second AI model also includes the neuron 1 and the neuron 2. Values of first parameters corresponding to the neurons 1 in all the second AI models are averaged, and then replace a value of the first parameter corresponding to the neuron 1 in the first AI model. Values of second parameters corresponding to the neurons 2 in all the second AI models are averaged, and then replace a value of the second parameter corresponding to the neuron 2 in the first AI model, to obtain a third AI model.

[00132] In addition to the foregoing restoration manner, a new restoration manner may be obtained through further variation. For example, for a same neuron in the second AI model corresponding to each client, an average value of parameters corresponding to the same neuron may be used, a product of the average value and a reference coefficient is used as a target parameter value, and a parameter value corresponding to the neuron in the first AI model is replaced with the target parameter value, to obtain a third AI model. That both the first AI model and the second AI model include a neuron 1 and a neuron 2 is still used as an example. Values of first parameters

corresponding to the neurons 1 in all the second AI models are averaged to obtain a first average value, a product of the first average value and a reference coefficient is used to replace a value of the first parameter corresponding to the neuron 1 in the first AI model, values of second parameters corresponding to the neurons 2 in all the second AI models are averaged to obtain a second average value, and a product of the second average value and the reference coefficient is used to replace a value of the second parameter corresponding to the neuron 2 in the first AI model, to obtain a third AI model. The reference coefficient may be set based on experience, or may be adjusted based on an application scenario. This is not limited in this embodiment of this application.

[00133] Regardless of which manner is used to perform federated averaging to obtain the third AI model, after the third AI model is obtained, it is determined whether the third AI model is converged, and in response to the fact that the third AI model is converged, the third AI model is used as the target AI model. Alternatively, in response to the fact that the third AI model is not converged, the third AI model is sent to the plurality of clients, a new AI model continues to be obtained in a manner of obtaining the third AI model, and this process is repeated until a converged target AI model is obtained.

[00134] A manner of determining whether the third AI model is converged is not limited in this embodiment of this application, and may be that a quantity of rounds of training reaches a training threshold, or that performance of the third AI model meets a specific requirement.

[00135] In conclusion, according to the method provided in this embodiment of this application, not all neurons in the model need to be trained, but some neurons are selected for training based on an active condition. This can reduce power consumption, and increase a training speed. In addition, because the client transmits only a parameter corresponding to a trained neuron to the service end, communication bandwidth can be reduced.

[00136] In addition, a frozen-active flag bit indicates the active condition, so that a target neuron can be directly determined by using the frozen-active flag bit. Therefore, a speed of determining the target neuron is relatively high. This further reduces training time and increases a training speed. A frozen period is set, a period in which a neuron does not participate in training is indicated based on the frozen period, and whether the neuron participates in training is determined based on the frozen period, thereby preventing loss of a parameter corresponding to a neuron that needs to participate in training in a training process, and ensuring robustness. A frozen period counter is set, and indicates whether the frozen period ends, and the activity degree is obtained after the frozen

period ends. This further reduces time for detecting whether the frozen period ends, and increases a training speed.

[00137] An overall procedure of the method for obtaining an artificial intelligence model provided in this embodiment of this application may be an interaction process shown in FIG. 4. The following describes the method for obtaining an artificial intelligence model provided in this embodiment of this application by using an interaction process between a service end and a target client in a plurality of clients as an example. As shown in FIG. 4, the method includes but is not limited to the following steps.

[00138] 401: The service end delivers a central model to the plurality of clients, and the target client receives the central model, where the central model has N neurons.

[00139] 402: The client starts local training. If the current round is a first round, initialization is performed on a frozen period Ti = 1, and i ∈ {1, 2, ..., N}, indicating that freeze detection is performed on the i^{th} neuron in the current round. Initialization is performed on a frozen-active flag bit Fi = 0, and i ∈ {1, 2, ..., N}, indicating that the i^{th} neuron is not frozen (if the value is 1, it indicates that the i^{th} neuron is frozen). Initialization is performed on a frozen period counter Tci = 1, and i ∈ {1, 2, ..., N} (Tci + 1 for each interaction with the service end). The current count of training based on a neuron freezing method is locally performed, to update the values Ti, Fi, and Tci.

[00140] 403: Each client uploads, to the service end, parameter data of a target neuron that participates in the current round of training.

[00141] 404: If the frozen-active flag bit Fi changes, the target client uploads the frozen-active flag bit Fi to the service end.

[00142] It should be noted that, for a neuron whose frozen-active flag bit Fi does not change, the frozen-active flag bit Fi of the neuron may not be transmitted. If the frozen-active flag bits Fi of all neurons do not change, the process of 403 may be omitted.

[00143] 405: The service end aggregates a local partial model of each client and the frozen-active flag bit, and restores each local model.

[00144] The service end aggregates the local partial model MI of each client, where I indicates a quantity of the clients. For example, a local partial model of a client 1 is M10, a local partial model of a client 2 is M20, a local partial model of a client 3 is M30, and the like.

[00145] For example, neurons are arranged from left to right and from top to bottom, and related

parameters of the neurons are arranged from top to bottom. A neuron with a frozen-active flag bit Fi = 1 is supplemented by using a nearest central model parameter, and a neuron with a frozen-active flag bit Fi = 0 is updated by using a local partial model parameter.

[00146] 406: The service end performs federated averaging on the local models, and updates the central model.

[00147] 407: The service end delivers the central model to each client, and then 401 is performed.

[00148] The foregoing process of 401 to 407 is repeated until the central model is converged to obtain a converged target AI model. In this case, the federated learning training ends.

[00149] A client performs the method for obtaining an artificial intelligence model. A model training process performed by the client is described by using an example shown in FIG. 5. As shown in FIG. 5, the method includes but is not limited to the following steps.

[00150] 501: The client receives a central model sent by a service end.

[00151] The central model is usually a known model trained offline, and a quantity of neurons is N. In addition, the client determines batches E of the current round of training. A batch of each round of training may be delivered by the service end to the client, or may be agreed by both parties. Usually, a quantity of batches of each round of training on each client is the same.

[00152] 502: The client initializes a frozen period, a frozen-active flag bit, and a frozen period counter of a neuron of the central model.

[00153] For example, the initialized frozen periods of neurons of the central model are all 1, namely, $T_i = 1, i \in \{1,2,\ldots,5\}$. If Ti = 1, the i^{th} neuron of a local model in the current round of training needs to perform a freeze detection mechanism to update the frozen period.

[00154] The initialized frozen-active flag bit of each neuron is 0, namely, $F_i = 0, i \in \{1,2,\ldots,5\}$. If Fi = 1, the i^{th} neuron of the local model in the current round of training is frozen, that is, a parameter corresponding to the neuron does not participate in training.

[00155] The initialized frozen period counter of each neuron is Tci = 1.

[00156] 503: The client queries a value Fi of each neuron based on the initial central model, freezes all neurons whose Fi is 1, and starts to train all neurons whose Fi is 0 for the batches E.

[00157] 504: The client queries a value Ti of each neuron, starts a freeze detection mechanism for all neurons whose Tci is equal to Ti, updates Ti and Fi, and perform reinitialization to obtain Tci = 1.

[00158] For all neurons whose Tci is not equal to Ti, Ti and Fi are maintained, and the counter Tci is updated to Tci + 1 in the frozen period.

[00159] 505: After the training ends, the client obtains a local model, and uploads the local model to the service end.

[00160] An embodiment of this application provides an apparatus for obtaining an artificial intelligence model. The apparatus is configured to perform the method performed by the client in the embodiment shown in FIG. 3. As shown in FIG. 6, the apparatus includes:

a communications unit 601, configured to receive a first AI model sent by a service end, where the first AI model includes a plurality of neurons, for example, a function performed by the communications unit 601 may be shown in 303 in FIG. 3, and details are not described herein again; and

a processing unit 602, configured to determine, from the plurality of neurons, a target neuron participating in a current round of training, where the current round of training is a non-first round of training, and a quantity of target neurons is less than a total quantity of the plurality of neurons; and for example, a function performed by the processing unit 602 may be shown in 303 in FIG. 3, and details are not described herein again.

[00161] The processing unit 602 is further configured to train the target neuron based on the local data, to obtain a parameter corresponding to the target neuron. For example, a function performed by the processing unit 602 may be shown in 305 in FIG. 3, and details are not described herein again.

[00162] The communications unit 601 is further configured to return parameter data corresponding to the target neuron to the service end. The parameter data corresponding to the target neuron is used by the service end to obtain a converged target AI model. For example, a function performed by the communications unit 601 may be shown in 306 in FIG. 3, and details are not described herein again.

[00163] In an example embodiment, each neuron in the plurality of neurons has a corresponding frozen-active flag bit, and when a value of the frozen-active flag bit is a first value, the frozen-active flag bit indicates that the neuron participates in the current round of training; and

the processing unit 602 is configured to determine the neuron whose value of frozen-active flag bit is the first value as the target neuron participating in the current round of training.

[00164] In an example embodiment, each neuron in the plurality of neurons further has a

corresponding frozen period, and the frozen period indicates a period in which the neuron does not participate in training; and

a value of a frozen-active flag bit corresponding to any neuron is determined based on a frozen period corresponding to the any neuron.

[00165] In an example embodiment, the processing unit 602 is further configured to: for the any neuron in the plurality of neurons, obtain an activity degree of the any neuron in a previous round of training, where the activity degree indicates a degree to which the neuron is affected by the local data; and update the frozen period of the any neuron based on the activity degree of the any neuron in the previous round of training, and update the value of the frozen-active flag bit of the any neuron based on ~~the~~ an updated frozen period of the any neuron.

[00166] In an example embodiment, each neuron in the plurality of neurons further has a corresponding frozen period counter, and the frozen period counter indicates whether the frozen period ends; and

the processing unit 602 is configured to: in response to the fact that a value of a frozen period counter of the any neuron indicates that the frozen period of the any neuron ends, obtain the activity degree of the any neuron in the previous round of training.

[00167] In an example embodiment, the processing unit 602 is configured to: in response to the fact that the current round of training is the non-first round of training, for any neuron in the plurality of neurons, obtain an activity degree of the any neuron in a previous round of training, where the activity degree indicates a degree to which the neuron is affected by the local data; and in response to the fact that the activity degree of the any neuron in the previous round of training is greater than an activity degree threshold, determine the any neuron as the target neuron participating in the current round of training.

[00168] In an example embodiment, the processing unit 602 is configured to: obtain a first average value of parameters obtained before the previous round of training of the any neuron and a second average value of parameters obtained after the previous round of training of the any neuron; obtain a difference between the first average value and the second average value; and determine the activity degree of the any neuron in the previous round of training based on an absolute value of the difference and an absolute value of the first average value.

[00169] In an example embodiment, the processing unit 602 is further configured to: receive the first artificial intelligence AI model sent by the service end, determine all neurons in the

plurality of neurons as target neurons, and train the target neurons based on the local data.

[00170] In an example embodiment, the communications unit 601 is further configured to send the frozen-active flag bit of each neuron to the service end~~server~~, or send, to the service end~~server~~, a frozen-active flag bit whose value changes.

[00171] In an example embodiment, the communications unit 601 is configured to return only the parameter data corresponding to the target neuron to the service end.

[00172] According to the apparatus provided in this embodiment of this application, not all neurons in the model need to be trained, but some neurons are selected for training based on an active condition. This can reduce power consumption, and increase a training speed. In addition, because the client transmits only a parameter corresponding to a trained neuron to the service end, communication bandwidth can be reduced.

[00173] In addition, a frozen-active flag bit indicates the active condition, so that a target neuron can be directly determined by using the frozen-active flag bit. Therefore, a speed of determining the target neuron is relatively high. This further reduces training time and increases the training speed. A frozen period is set, a period in which a neuron does not participate in training is indicated based on the frozen period, and whether the neuron participates in training is determined based on the frozen period, thereby preventing loss of a parameter corresponding to a neuron that needs to participate in training in a training process, and ensuring robustness. A frozen period counter is set, and indicates whether the frozen period ends, and the activity degree is obtained after the frozen period ends. This further reduces time for detecting whether the frozen period ends, and increases the training speed.

[00174] An embodiment of this application provides an apparatus for obtaining an artificial intelligence model. The apparatus is configured to perform a function performed by the service end in the embodiment shown in FIG. 3. As shown in FIG. 7, the apparatus includes:

a processing unit 701, configured to obtain a to-be-trained first artificial intelligence AI model, where the first AI model includes a plurality of neurons, for example, a function performed by the processing unit 701 may be shown in 301 in FIG. 3, and details are not described herein again; and

a communications unit 702, configured to send the first AI model to a plurality of clients, for example, a function performed by the communications unit 702 may be shown in 302 in FIG. 3, and details are not described herein again.

[00175] The communications unit 702 is further configured to receive parameter data that corresponds to a target neuron and that is returned by each client in the plurality of clients, where parameter data that corresponds to the target neuron and that is returned by any client is obtained by the any client by training the target neuron in the first AI model, and a quantity of target neurons is less than a total quantity of the plurality of neurons. For example, a function performed by the communications unit 702 may be shown in 307 in FIG. 3, and details are not described herein again.

[00176] The processing unit 701 is further configured to: restore, based on the parameter data that corresponds to the target neuron and that is returned by each client, a second AI model corresponding to each client, and obtain a converged target AI model based on the second AI model corresponding to each client. For example, a function performed by the processing unit 701 may be shown in 307 in FIG. 3, and details are not described herein again.

[00177] In an example embodiment, the processing unit 701 is configured to: for any client, update, based on a parameter that corresponds to the target neuron and that is returned by the any client, a parameter corresponding to the target neuron in the first AI model, and supplement a parameter corresponding to a non-target neuron in the first AI model, to obtain a second AI model corresponding to the any client.

[00178] In an example embodiment, the processing unit 701 is configured to: for any client, update, based on the parameter data that corresponds to the target neuron and that is returned by the any client, a parameter corresponding to the target neuron in the first AI model, and supplement a parameter corresponding to a non-target neuron in the first AI model, to obtain a second AI model corresponding to the any client.

[00179] In an example embodiment, the processing unit 701 is configured to: perform federated averaging on the second AI model corresponding to each client, to obtain a third AI model; and in response to the fact that the third AI model is converged, use the third AI model as the target AI model; or in response to the fact that the third AI model is not converged, send the third AI model to the plurality of clients, continue to obtain a new AI model in a manner of obtaining the third AI model, and repeat this process until the converged target AI model is obtained.

[00180] In an example embodiment, the communications unit 702 is further configured to receive a frozen-active flag bit of each neuron or a frozen-active flag bit whose value changes, that is returned by the plurality of clients.

[00181] The processing unit 701 is configured to determine the target neuron in the plurality of neurons by using the frozen-active flag bit of each neuron or the frozen-active flag bit whose value changes.

[00182] According to the apparatus provided in this embodiment of this application, the client does not train all neurons in the model, but selects some neurons for training based on an active condition, and uploads only a parameter corresponding to a trained target neuron. Therefore, communication bandwidth can be reduced, and a training speed can be increased. In this way, a speed of obtaining the target AI model is increased.

[00183] Furthermore, in addition to updating the parameter corresponding to the target neuron in the first AI model, a parameter corresponding to a non-target neuron in the first AI model is also supplemented. This ensures integrity of the model, so that performance of the obtained target AI model is better.

[00184] It should be understood that, when the apparatus provided in FIG. 6 or FIG. 7 implements functions of the apparatus, division into the functional modules is merely used as an example for description. During actual application, the foregoing functions may be allocated to different functional modules for implementation based on a requirement. In other words, an inner structure of a device is divided into different functional modules, to implement all or some of the functions described above. In addition, the apparatuses provided in the foregoing embodiments and the method embodiments pertain to a same concept. For a specific implementation process of the apparatuses, refer to the method embodiments. Details are not described herein again.

[00185] FIG. 8 is a schematic diagram of a hardware structure of a device 1100 for obtaining an artificial intelligence model according to an embodiment of this application. The device 1100 for obtaining an artificial intelligence model shown in FIG. 8 may perform corresponding steps performed by the service end or the client in the methods in the foregoing embodiments.

[00186] As shown in FIG. 8, the device 1100 for obtaining an artificial intelligence model includes a processor 1101, a memory 1102, an interface 1103, and a bus 1104. The interface 1103 may be implemented in a wireless or wired manner, and may be specifically a network adapter. The processor 1101, the memory 1102, and the interface 1103 are connected through the bus 1104.

[00187] The interface 1103 may include a transmitter and a receiver. The device for obtaining an artificial intelligence model may receive and send, through the interface 1103, a parameter corresponding to a target neuron.

[00188] For example, the device 1100 for obtaining an artificial intelligence model shown in FIG. 8 is the client in FIG. 3, and the ~~processor 1002~~processor 1101 reads instructions in the ~~memory 1001~~memory 1102, so that the device 1100 for obtaining an artificial intelligence model shown in FIG. 8 can perform all or some operations performed by the client.

[00189] For another example, the device 1100 for obtaining an artificial intelligence model shown in FIG. 8 is the service end in FIG. 3, and the ~~processor 1002~~processor 1101 reads instructions in the ~~memory 1001~~memory 1102, so that the device 1100 for obtaining an artificial intelligence model shown in FIG. 8 can perform all or some operations performed by the service end.

[00190] In addition, the processor 1101 is configured to perform another process of the technology described in this specification. The memory 1102 includes an operating system 11021 and an application 11022, and is configured to store a program, code, or instructions. When executing the program, the code, or the instructions, the processor or a hardware device may complete a processing process related to the device 1100 for obtaining an artificial intelligence model in the method embodiment. Optionally, the memory 1102 may include a read-only memory (English: Read-only Memory, ROM for short) and a random access memory (English: Random Access Memory, RAM for short). The ROM includes a basic input/output system (English: Basic Input/Output System, BIOS for short) or an embedded system. The RAM includes an application program and an operating system. When the device 1100 for obtaining an artificial intelligence model needs to be run, a system is booted by using a BIOS fixed in a ROM or a bootloader in an embedded system, and the device 1100 for obtaining an artificial intelligence model is booted into a normal running state. After entering the normal running state, the device 1100 for obtaining an artificial intelligence model runs the application and the operating system in the RAM, to complete a processing process related to the device 1100 for obtaining an artificial intelligence model in the method embodiments.

[00191] It may be understood that FIG. 8 shows only a simplified design of the device 1100 for obtaining an artificial intelligence model. During actual application, the device 1100 for obtaining an artificial intelligence model may include any quantity of interfaces, processors, or memories.

[00192] It should be understood that the processor may be a central processing unit (Central Processing Unit, CPU), or may be another general-purpose processor, a digital signal processor (digital signal processor, DSP), an application-specific integrated circuit (application-specific

integrated circuit, ASIC), a field-programmable gate array (field-programmable gate array, FPGA) or another programmable logic device, a discrete gate or a transistor logic device, a discrete hardware component, or the like. The general-purpose processor may be a microprocessor, any conventional processor, or the like. It should be noted that the processor may be a processor that supports an advanced reduced instruction set computer **Error! Hyperlink reference not valid.**machines (advanced RISC machines, ARM) architecture.

[00193] Further, in an optional embodiment, the memory may include a read-only memory and a random access memory, and provide instructions and data for the processor. The memory may further include a nonvolatile random access memory. For example, the memory may further store information of a device type.

[00194] The memory may be a volatile memory or a nonvolatile memory, or may include both a volatile memory and a nonvolatile memory. The nonvolatile memory may be a read-only memory (read-only memory, ROM), a programmable read-only memory (programmable ROM, PROM), an erasable programmable read-only memory (erasable PROM, EPROM), an electrically erasable programmable read-only memory (electrically EPROM, EEPROM), or a flash memory. The volatile memory may be a random access memory (random access memory, RAM), used as an external cache. By way of example but not limitation, many forms of RAMs may be used, for example, a static random access memory (static RAM, SRAM), a dynamic random access memory (dynamic random access memory, DRAM), a synchronous dynamic random access memory (synchronous DRAM, SDRAM), a double data rate synchronous dynamic random access memory (double data rate SDRAM, DDR SDRAM), an enhanced synchronous dynamic random access memory (enhanced SDRAM, ESDRAM), a synchlink dynamic random access memory (synchlink DRAM, SLDRAM), and a direct rambus random access memory (direct rambus RAM, DR RAM).

[00195] A computer-readable storage medium is further provided. The storage medium stores at least one instruction, and the instruction is loaded and executed by a processor to implement any one of the foregoing methods for obtaining an artificial intelligence model.

[00196] This application provides a computer program. When the computer program is executed by a computer, a processor or the computer is enabled to perform corresponding steps and/or procedures in the foregoing method embodiments.

[00197] A chip is provided. The chip includes a processor, configured to: invoke, from a memory, instructions stored in the memory and run the instructions, so that a communications

device on which the chip is installed performs the methods in the foregoing aspects.

[00198] Another chip is provided. The chip includes an input interface, an output interface, a processor, and a memory. The input interface, the output interface, the processor, and the memory are connected to each other through an internal connection channel. The processor is configured to execute code in the memory. When the code is executed, the processor is configured to perform the methods in the foregoing aspects.

[00199] All or some of the foregoing embodiments may be implemented by using software, hardware, firmware, or any combination thereof. When the software is used to implement the embodiments, all or some of the embodiments may be implemented in a form of a computer program product. The computer program product includes one or more computer instructions. When the computer program instructions are loaded and executed on a computer, the procedure or functions according to this application are all or partially generated. The computer may be a general-purpose computer, a dedicated computer, a computer network, or another programmable apparatus. The computer instructions may be stored in the computer-readable storage medium or may be transmitted from a computer-readable storage medium to another computer-readable storage medium. For example, the computer instructions may be transmitted from a website, computer, server, or data center to another website, computer, server, or data center in a wired (for example, a coaxial cable, an optical fiber, or a digital subscriber line) or wireless (for example, infrared, radio, or microwave) manner. The computer-readable storage medium may be any usable medium accessible by the computer, or a data storage device, such as a server or a data center, integrating one or more usable media. The usable medium may be a magnetic medium (for example, a floppy disk, a hard disk, or a magnetic tape), an optical medium (for example, a DVD), or a semiconductor medium (for example, a solid state disk Solid State Disk).

[00200] In the foregoing specific implementations, the objectives, technical solutions, and beneficial effects of this application are further described in detail. It should be understood that the foregoing descriptions are merely specific implementations of this application, but are not intended to limit the protection scope of this application. Any modification, equivalent replacement, improvement, or the like made based on the technical solutions of this application shall fall within the protection scope of this application.

CLAIMS

What is claimed is:

1. A method for obtaining an artificial intelligence model, wherein the method comprises:

receiving, by a client, a first artificial intelligence (AI) model sent by a service end, wherein the first AI model comprises a plurality of neurons;

determining, by the client from the plurality of neurons, a target neuron participating in a current round of training, wherein the current round of training is a non-first round of training, and a quantity of target neurons is less than a total quantity of the plurality of neurons;

training, by the client, the target neuron based on local data; and

returning, by the client, parameter data corresponding to the target neuron to the service end, wherein the parameter data corresponding to the target neuron is used by the service end to obtain a converged target AI model.

2. The method according to claim 1, wherein each neuron in the plurality of neurons has a corresponding frozen-active flag bit, and when a value of the frozen-active flag bit is a first value, the frozen-active flag bit indicates that the neuron participates in the current round of training; and

the determining, from the plurality of neurons, a target neuron participating in a current round of training comprises:

determining the neuron whose value of frozen-active flag bit is the first value as the target neuron participating in the current round of training.

3. The method according to claim 2, wherein each neuron in the plurality of neurons has a corresponding frozen period, and the frozen period indicates a period in which the neuron does not participate in training; and

a value of a frozen-active flag bit corresponding to any neuron is determined based on a frozen period corresponding to the any neuron.

4. The method according to claim 3, wherein before the determining the neuron whose value of frozen-active flag bit is the first value as the target neuron participating in the current round of training, the method further comprises:

for the any neuron in the plurality of neurons, obtaining an activity degree of the any neuron in a previous round of training, wherein the activity degree indicates a degree to which the neuron is affected by the local data; and

updating the frozen period of the any neuron based on the activity degree of the any neuron in the previous round of training, and updating the value of the frozen-active flag bit of the any neuron based on ~~the~~ an updated frozen period of the any neuron.

5. The method according to claim 4, wherein each neuron in the plurality of neurons further has a corresponding frozen period counter, and the frozen period counter indicates whether the frozen period ends; and

the obtaining an activity degree of the any neuron in a previous round of training comprises:

in response to the fact that a value of a frozen period counter of the any neuron indicates that the frozen period of the any neuron ends, obtaining the activity degree of the any neuron in the previous round of training.

6. The method according to claim 1, wherein the determining, from the plurality of neurons, a target neuron participating in a current round of training comprises:

for any neuron in the plurality of neurons, obtaining an activity degree of the any neuron in a previous round of training, wherein the activity degree indicates a degree to which the neuron is affected by the local data; and

in response to the fact that the activity degree of the any neuron in the previous round of training is greater than an activity degree threshold, determining the any neuron as the target neuron participating in the current round of training.

7. The method according to any one of claims 4 to 6, wherein the obtaining an activity degree of the any neuron in a previous round of training comprises:

obtaining a first average value of parameters obtained before the previous round of training of the any neuron and a second average value of parameters obtained after the previous round of training of the any neuron;

obtaining a difference between the first average value and the second average value; and

determining the activity degree of the any neuron in the previous round of training based on an absolute value of the difference and an absolute value of the first average value.

~~8. The method according to claim 1, wherein before the receiving a first artificial intelligence AI model sent by a service end, the method further comprises:~~

~~receiving the first artificial intelligence AI model sent by the service end, determining all neurons in the plurality of neurons as target neurons, and training the target neurons based on the local data.~~

~~9~~8. The method according to any one of claims 2 to 5, wherein the returning parameter data corresponding to the target neuron to the service end further comprises:

sending the frozen-active flag bit of each neuron to the service end~~server~~, or sending, to the service end~~server~~, a frozen-active flag bit whose value changes.

~~10~~9. The method according to any one of claims 1 to ~~9~~8, wherein the returning parameter data corresponding to the target neuron to the service end comprises: returning only the parameter data corresponding to the target neuron to the service end.

~~11~~10. A method for obtaining an artificial intelligence model, wherein the method comprises:

obtaining, by a service end, a to-be-trained first artificial intelligence (AI) model, wherein the first AI model comprises a plurality of neurons;

sending, by the service end, the first AI model to a plurality of clients;

receiving, by the service end, parameter data that corresponds to a target neuron and that is returned by each client in the plurality of clients, wherein parameter data that corresponds to the target neuron and that is returned by any client is obtained by the any client by training the target neuron in the first AI model, and a quantity of target neurons is less than a total quantity of the plurality of neurons; and

restoring, by the service end based on the parameter data that corresponds to the target neuron and that is returned by each client, a second AI model corresponding to each client, and obtaining a converged target AI model based on the second AI model corresponding to each client.

~~12~~11. The method according to claim ~~11~~10, wherein the restoring, based on the parameter data that corresponds to the target neuron and that is returned by each client, a second AI model corresponding to each client comprises:

for the any client, updating, based on the parameter data that corresponds to the target neuron and that is returned by the any client, a parameter corresponding to the target neuron in the first AI model, to obtain a second AI model corresponding to the any client.

~~13. The method according to claim 11 or 12, wherein the obtaining a converged target AI model based on the second AI model corresponding to each client comprises:~~

~~performing federated averaging on the second AI model corresponding to each client, to obtain a third AI model; and~~

~~in response to the fact that the third AI model is converged, using the third AI model as the target AI model; or~~

~~in response to the fact that the third AI model is not converged, sending the third AI model to the plurality of clients, continuing to obtain a new AI model in a manner of obtaining the third AI model, and repeating this process until the converged target AI model is obtained.~~

1~~4~~2. The method according to claim 10 or 11~~any one of claims 10 to 13~~, wherein before the restoring, based on the parameter data that corresponds to the target neuron and that is returned by each client, a second AI model corresponding to each client, the method further comprises:

receiving a frozen-active flag bit of each neuron or a frozen-active flag bit whose value changes, that is returned by the plurality of clients; and

determining the target neuron in the plurality of neurons by using the frozen-active flag bit of each neuron or the frozen-active flag bit whose value changes.

~~15~~13. An apparatus for obtaining an artificial intelligence model, wherein the apparatus comprises:

a communications unit, configured to receive a first artificial intelligence (AI) model sent by a service end, wherein the first AI model comprises a plurality of neurons; and

a processing unit, configured to determine, from the plurality of neurons, a target neuron participating in a current round of training, wherein the current round of training is a non-first round of training, and a quantity of target neurons is less than a total quantity of the plurality of neurons;

the processing unit is further configured to train the target neuron based on local data; and

the communications unit is further configured to return parameter data corresponding to the target neuron to the service end, wherein the parameter data corresponding to the target neuron is used by the service end to obtain a converged target AI model.

~~16~~14. The apparatus according to claim ~~15~~13, wherein each neuron in the plurality of neurons has a corresponding frozen-active flag bit, and when a value of the frozen-active flag bit is a first value, the frozen-active flag bit indicates that the neuron participates in the current round of training; and

the processing unit is configured to determine the neuron whose value of frozen-active flag bit is the first value as the target neuron participating in the current round of training.

~~17~~15. The apparatus according to claim ~~16~~14, wherein each neuron in the plurality of neurons further has a corresponding frozen period, and the frozen period indicates a period in which the neuron does not participate in training; and

a value of a frozen-active flag bit corresponding to any neuron is determined based on a frozen period corresponding to the any neuron.

~~18~~16. The apparatus according to claim ~~17~~15, wherein the processing unit is further configured to: for the any neuron in the plurality of neurons, obtain an activity degree of the any neuron in a previous round of training, wherein the activity degree indicates a degree to which the neuron is affected by the local data; and update the frozen period of the any neuron based on the activity degree of the any neuron in the previous round of training, and update the value of the frozen-active flag bit of the any neuron based on ~~the~~ an updated frozen period of the any neuron.

~~19~~17. The apparatus according to claim ~~18~~16, wherein each neuron in the plurality of neurons further has a corresponding frozen period counter, and the frozen period counter indicates whether the frozen period ends; and

the processing unit is configured to: in response to the fact that a value of a frozen period counter of the any neuron indicates that the frozen period of the any neuron ends, obtain the activity degree of the any neuron in the previous round of training.

~~20~~18. The apparatus according to claim ~~15~~13, wherein the processing unit is configured to: in response to the fact that the current round of training is the non-first round of training, for any neuron in the plurality of neurons, obtain an activity degree of the any neuron in a previous round of training, wherein the activity degree indicates a degree to which the neuron is affected by the local data; and in response to the fact that the activity degree of the any neuron in the previous round of training is greater than an activity degree threshold, determine the any neuron as the target neuron participating in the current round of training.

~~21~~19. The apparatus according to any one of claims ~~18~~ 16 to ~~20~~18, wherein the processing unit is configured to: obtain a first average value of parameters obtained before the previous round of training of the any neuron and a second average value of parameters obtained after the previous round of training of the any neuron; obtain a difference between the first average value and the second average value; and determine the activity degree of the any neuron in the previous round of training based on an absolute value of the difference and an absolute value of the first average value.

~~22. The apparatus according to claim 15, wherein the processing unit is further configured to: receive the first artificial intelligence AI model sent by the service end, determine all neurons in the plurality of neurons as target neurons, and train the target neurons based on the local data.~~

2330. The apparatus according to any one of claims 16–14 to 1917, wherein the communications unit is further configured to send the frozen-active flag bit of each neuron to the service endserver, or send, to the service endserver, a frozen-active flag bit whose value changes.

2421. The apparatus according to any one of claims 15–13 to 2320, wherein the communications unit is configured to return only the parameter data corresponding to the target neuron to the service end.

2522. An apparatus for obtaining an artificial intelligence model, wherein the apparatus comprises:

a processing unit, configured to obtain a to-be-trained first artificial intelligence (AI) model, wherein the first AI model comprises a plurality of neurons; and

a communications unit, configured to send the first AI model to a plurality of clients;

the communications unit is further configured to receive parameter data that corresponds to a target neuron and that is returned by each client in the plurality of clients, wherein parameter data that corresponds to the target neuron and that is returned by any client is obtained by the any client by training the target neuron in the first AI model, and a quantity of target neurons is less than a total quantity of the plurality of neurons; and

the processing unit is further configured to: restore, based on the parameter data that corresponds to the target neuron and that is returned by each client, a second AI model corresponding to each client, and obtain a converged target AI model based on the second AI model corresponding to each client.

2623. The apparatus according to claim 2522, wherein the processing unit is configured to: for any client, update, based on the parameter data that corresponds to the target neuron and that is returned by the any client, a parameter corresponding to the target neuron in the first AI model, and supplement a parameter corresponding to a non-target neuron in the first AI model, to obtain a second AI model corresponding to the any client.

2724. The apparatus according to claim 25–22 or 2623, wherein the processing unit is configured to: perform federated averaging on the second AI model corresponding to each client, to obtain a third AI model; and in response to the fact that the third AI model is converged, use the third AI model as the target AI model; or in response to the fact that the third AI model is not converged, send the third AI model to the plurality of clients, continue to obtain a new AI model in a manner of obtaining the third AI model, and repeat this process until the converged target AI

model is obtained.

~~28~~25. The apparatus according to any one of claims ~~24~~ 21 to ~~27~~24, wherein the communications unit is further configured to receive a frozen-active flag bit of each neuron or a frozen-active flag bit whose value changes, that is returned by the plurality of clients; and

the processing unit is configured to determine the target neuron in the plurality of neurons by using the frozen-active flag bit of each neuron or the frozen-active flag bit whose value changes.

~~29~~26. A device for obtaining an artificial intelligence model, wherein the device comprises:

a memory and a processor, wherein the memory stores at least one instruction, and the at least one instruction is loaded and executed by the processor, to implement the method for obtaining an artificial intelligence model according to any one of claims 1 to ~~10~~9.

~~30~~27. A device for obtaining an artificial intelligence model, wherein the device comprises:

a memory and a processor, wherein the memory stores at least one instruction, and the at least one instruction is loaded and executed by the processor, to implement the method for obtaining an artificial intelligence model according to any one of claims ~~11~~ 10 to ~~14~~12.

~~31~~28. A system for obtaining an artificial intelligence model, wherein the system comprises the device according to claim ~~29~~ 26 and a plurality of devices according to claim ~~30~~27.

~~32~~29. A computer-readable storage medium, wherein the storage medium stores at least one instruction, and the instruction is located and executed by a processor to implement the method for obtaining an artificial intelligence model according to any one of claims 1 to ~~14~~12.