DATA PROCESSING METHODS AND APPARATUSES, AND DEVICES

TECHNICAL FIELD

[0001] This application relates to the field of artificial intelligence, and in particular to a data processing method and apparatus, and a device.

BACKGROUND

[0002] Machine learning is an approach to artificial intelligence (AI), and is a multi-field interdisciplinary subject involving probability theory, statistics, approximation theory, convex analysis, algorithm complexity theory, and many other disciplines. Machine learning is used to study how computers simulate or implement human learning behaviors to acquire new knowledge or skills, and to reorganize existing knowledge structures to continuously improve their own performance. Machine learning pays more attention to algorithm design, enabling computers to automatically learn patterns from data and use the patterns to predict unknown data.

[0003] Machine learning has been used in a wide variety of applications, such as deep learning, data mining, computer vision, natural language processing, biometric recognition, search engines, medical diagnosis, detection of credit card fraud, securities market analysis, DNA sequence sequencing, speech and handwriting recognition, strategy games, and robotics.

SUMMARY

[0004] The present application provides a data processing method, including:

[0005] transmitting, by a central server, an initial baseline model to an edge server;

[0006] training, by the edge server, the initial baseline model with scenario data from the edge server to obtain a target baseline model, and determining whether to deploy the target baseline model or not;

[0007] transmitting, by the edge server, the target baseline model to the central server if it is determined that the target baseline model is not deployed; and

[0008] generating, by the central server, a fusion baseline model based on the target baseline model and the initial baseline model, training the fusion baseline model, determining the trained baseline model as an initial baseline model, and returning to an operation of transmitting, by the central server, the initial baseline model to the edge server.

[0009] The present application provides a data processing method, applied to an edge server, the method including:

[0010] acquiring an initial baseline model from a central server;

[0011] training the initial baseline model with scenario data from the edge server to obtain a target baseline model, and determining whether to deploy the target baseline model or not; and

[0012] if it is determined that the target baseline model is not deployed, transmitting the target baseline model to the central server, such that the central server generates a fusion baseline model based on the target baseline model and the initial baseline model, and reacquires an initial baseline model based on the fusion baseline model, and returning to an operation of acquiring the initial baseline model from the central server.

[0013] The present application provides a data processing method, applied to a central server, the method including:

[0014] transmitting an initial baseline model to an edge server, such that the edge server trains the initial baseline model with scenario data from the edge server to obtain a target baseline model;

[0015] acquiring the target baseline model from the edge server;

[0016] generating a fusion baseline model based on the target baseline model and the initial baseline model; and

[0017] training the fusion baseline model, determining the trained baseline model as an initial baseline model, and returning to an operation of transmitting the initial baseline model to the edge server.

[0018] The present application provides a data processing apparatus, applied to an edge server, the apparatus including:

[0019] an acquiring module, configured to acquire an initial baseline model from a central server;

[0020] a training module, configured to train the initial baseline model with scenario data from the edge server to obtain a target baseline model, and determine whether to deploy the target baseline model or not; and

[0021] a transmitting module, configured to transmit the target baseline model to the central server if it is determined that the target baseline model is not deployed, such that the central server generates a fusion baseline model based on the target baseline model and the initial baseline model, and reacquires an initial baseline model based on the fusion baseline model.

[0022] The present application provides a data processing apparatus, applied to a central server, the apparatus including:

[0023] a transmitting module, configured to transmit an initial baseline model to an edge server, such that the edge server trains the initial baseline model with scenario data from the edge server to obtain a target baseline model;

[0024] an acquiring module, configured to acquire the target baseline model from the edge server; and

[0025] a generating module, configured to generate a fusion baseline model based on the target baseline model and the initial baseline model, train the fusion baseline model, and determine the trained baseline model as an initial baseline model.

[0026] The present application provides an edge server, including a processor and a machine-readable storage medium storing machine-executable instructions executable by the processor,

[0027] where the processor is configured to execute the machine-executable instructions to implement operations including:

[0028] acquiring an initial baseline model from a central server;

[0029] training the initial baseline model with scenario data from the edge server to obtain a target baseline model, and determining whether to deploy the target baseline model or not; and

[0030] transmitting the target baseline model to the central server if it is determined that the target baseline model is not deployed, such that the central server generates a fusion baseline model based on the target baseline model and the initial baseline model, and reacquires an initial baseline model based on the fusion baseline model, and returning to the operation of acquiring the initial baseline model from the central server.

[0031] The present application provides a central server, including a processor and a machine-readable storage medium storing machine-executable instructions executable by the processor,

[0032] where the processor is configured to execute the machine-executable instructions to implement operations including:

[0033] transmitting an initial baseline model to an edge server, such that the edge server trains the initial baseline model with scenario data from the edge server to obtain a target baseline model;

[0034] acquiring the target baseline model from the edge server;

[0035] generating a fusion baseline model based on the target baseline model and the initial baseline model; and

[0036] training the fusion baseline model, determining the trained baseline model as an initial baseline model, and returning to the operation of transmitting the initial baseline model to the edge server.

[0037] As above, in embodiments of the present application, an edge server may acquire an initial baseline model from a central server, train the initial baseline model with scenario data to obtain a target baseline model, and transmit the target baseline model to the central server, and the central server generates a fusion baseline model based on the target baseline model and the initial baseline model, trains the fusion baseline model, determines the trained baseline model as an initial baseline model, and transmits the initial baseline model to the edge server, and so on. Since the above process may be performed cyclically, the performance of the initial baseline model and the target baseline model is continuously upgraded, and the recognition capability of the initial baseline model and the target baseline model is continuously improved, such that the target baseline model achieves the expected performance and high-accuracy recognition capability. After several iterations, the edge server may obtain the target baseline model with better performance, the target baseline model can match the environment where the edge server is located, and the intelligent analysis results have higher accuracy. In the above process, the edge server transmits the target baseline model to the central server instead of the scenario data (e.g., license plate images), so as to protect the privacy of the scenario data and not transmit the scenario data to the central server, thus realizing the privacy-removal data protection function, and avoiding transmitting the private license plate images to the central server. Since the target baseline model is trained with the scenario data, the information of the scenario data can be reflected in the initial baseline model and the target baseline model.

BRIEF DESCRIPTION OF THE DRAWINGS

[0038] In order to explain embodiments of the present application more clearly, the drawings to be used in the description of the embodiments of the present application will be briefly introduced below. It is apparent that the drawings in the following description only illustrate some embodiments described in the present application. For those ordinary skilled in the art, other drawings may also be obtained from these drawings in the embodiments of the present application.

[0039] FIG. 1 is a schematic structural diagram of a system according to an embodiment of the present application.

[0040] FIG. 2 is a schematic flowchart of a data processing method according to an embodiment of the present application.

[0041] FIG. 3 is a schematic structural diagram of a baseline model according to an embodiment of the present application.

[0042] FIG. 4 is a schematic flowchart of a data processing method according to another embodiment of the present application.

[0043] FIG. 5 is a schematic flowchart of a data processing method according to another embodiment of the present application.

[0044] FIGS. 6A and 6B are schematic structural diagrams of a data processing apparatus according to an embodiment of the present application.

[0045] FIG. 7A is a diagram illustrating a hardware structure of an edge server according to an embodiment of the present application.

[0046] FIG. 7B is a diagram illustrating a hardware structure of a central server according to an embodiment of the present application.

DETAILED DESCRIPTION OF THE EMBODIMENTS

[0047] Terms used in embodiments of the present application are only for the purpose of describing specific embodiments, rather than limiting the present application. As used herein, singular forms of “a”, “said”, and “the” are intended to include plural forms as well, unless the context clearly indicates otherwise. It should also be understood that the term “and/or” as used herein refers to including any or all possible combinations of one or more of the associated listed items.

[0048] It should be understood that, though terms “first”, “second”, “third”, etc. may be used in the embodiments of the present application to describe various information, the information should not be limited by these terms. These terms are only used to distinguish the same type of information from each other. For example, without departing from the scope of the present application, first information may also be referred to as second information, and similarly, second information may also be referred to as first information. Furthermore, depending on the context, the word “if” as used may be interpreted as “when”, “upon”, or “in response to determining”.

[0049] Before presenting embodiments of the present application, concepts related to the embodiments of the present application will be introduced first.

[0050] **Machine Learning**: Machine learning is an approach to artificial intelligence (AI), and is used to study how computers simulate or implement human learning behaviors to acquire new knowledge or skills, and to reorganize existing knowledge structures to continuously improve their own performance. Deep learning, a subcategory of machine learning, is a process in which mathematical models are used to model specific problems in the real world to solve similar problems in the field. Neural network is an implementation of deep learning. For the convenience of description, this specification takes the neural network as an example to introduce the structure and function of the neural network. For other subcategories of machine learning, the structure and function are similar to those of the neural network.

[0051] **Neural Network**: Neural network may include, but is not limited to, convolutional neural network (CNN for short), recurrent neural network (RNN for short), fully-connected network, etc. A structural unit of the neural network may include, but is not limited to, convolutional layer (Conv), pooling layer (Pool), activation layer, fully-connected layer (FC), etc., which is not limited.

[0052] In practical applications, one or more convolutional layers, one or more pooling layers, one or more activation layers, and one or more fully-connected layers may be combined to construct a neural network according to different requirements.

[0053] In the convolutional layer, input data features are enhanced by using a convolution kernel to perform a convolution operation on the input data features. The convolution kernel may be an m*n matrix, and the input data features of the convolutional layer are convolved with the convolution kernel to obtain output data features of the convolutional layer. The convolution operation is actually a filtering process.

[0054] In the pooling layer, by performing operations on input data features (e.g., the output of the convolutional layer) such as taking the maximum value, taking the minimum value, and taking the average value of the input data features, the input data features are sub-sampled using the principle of local correlation, reducing the amount of processing and maintaining feature invariance. The pooling operation is actually a downsampling process.

[0055] In the activation layer, an activation function (e.g., a nonlinear function) may be used to map input data features, thereby introducing a nonlinear factor that allows the neural network to enhance the expressive power through a nonlinear combination. The activation function may include, but is not limited to, a ReLU (Rectified Linear Units) function, which is configured to set features less than 0 to 0, while keeping features greater than 0 unchanged.

[0056] In the fully-connected layer, all data features input to this fully-connected layer are fully connected to obtain a feature vector, and the feature vector may include a plurality of data features.

[0057] **Baseline Model of the Neural Network** (e.g., the convolutional neural network): During the training of the neural network, various parameters in the neural network, such as, but not limited to, parameters of the convolutional layer (e.g., the convolution kernel), parameters of the pooling layer, parameters of the activation layer, and parameters of the fully-connected layer, may be trained using sample data. By training various parameters in the neural network, the neural network may be enabled to establish the mapping relationship between input and output.

[0058] After the training of the neural network is completed, the trained neural network is the baseline model of the neural network, which is referred to as "baseline model" herein. Artificial intelligence processing, such as face detection, human detection, vehicle detection, and license plate recognition, may be implemented based on the baseline model. For example, for an application scenario of license plate recognition, images involving license plates (that is, license plate images) may be input to the baseline model, the artificial intelligence processing is performed on the images through the baseline model, and the result for the artificial intelligence processing is the license plate recognition result.

[0059] **Sample Data and Scenario Data**: In an AI scenario, a central server and an edge server may be deployed. The central server acquires data used to train the baseline model and trains the baseline model with these data, and the data used by the central server to train the baseline model is called sample data. The edge server acquires data used to train the baseline model (which will not be transmitted to the central server) and trains the baseline model with these data, and the data used by the edge server to train the baseline model is called scenario data.

[0060] Exemplarily, the sample data may be image data or other types of data, which is not limited. The scenario data may be image data or other types of data, which is not limited.

[0061] Exemplarily, when the AI scenario is an AI scenario in the field of intelligent transportation, the above baseline model may be a baseline model for license plate recognition. On this basis, the sample data may be license plate images (that is, images involving license plate information) acquired by the central server to train the baseline model, and the central server may train the baseline model with the license plate images. However, the license plate image is just an example of the sample data, which is not limited.

The scenario data may be license plate images acquired by the edge server to train the baseline model, which will not be transmitted to the central server, and the edge server may train the baseline model with the license plate images. However, the license plate image is just an example of the scenario data, which is not limited.

[0062] As above, since the edge server will not transmit the scenario data to the central server, the central server may not train the baseline model with the scenario data from the edge server, that is, the baseline model may not match the environment where the edge server is located, and the baseline model has a poor performance after deployed to the edge server. Exemplarily, in the field of intelligent transportation, the central server may use license plate images as the sample data to train a baseline model for license plate recognition. However, in practical applications, the license plate images are private data, and based on the requirements for protection of private data, the edge server will not transmit the license plate images to the central server after acquiring the license plate images, resulting in the central server being unable to train the baseline model with the license plate images from the edge server. That is, the baseline model may not match the environment where the edge server is located, and the baseline model has a poor performance after deployed to the edge server.

[0063] Considering this, in the embodiments of the present application, after the baseline model is deployed to the edge server, the edge server may train the baseline model by using the scenario data of the environment where the edge server is located to obtain a new baseline model. Since the baseline model is trained by using the scenario data of the environment where the edge server is located, the new baseline model may match the environment where the edge server is located, and has a better performance.

[0064] The present application will be described below in conjunction with specific embodiments.

[0065] FIG. 1 is a schematic structural diagram of a system according to an embodiment of the present application. Referring to FIG. 1, the system may include a central server 110 and an edge server 120, a first relay device 130 connected with the central server 110, and a second relay device 140 connected with the edge server 120, the first relay device 130 being connected with the second relay device 140 via a network. The number of the edge server 120 is at least one, and each edge server 120 is connected with a second relay device 140, that is, the number of the second relay device 140 is the same as the number of the edge server 120.

[0066] The central server 110 may form a central recognition system, the at least one edge server 120 may form an end-side recognition system, and the first relay device 130 and the at least one second relay device 140 may form a privacy-removal system.

[0067] The central server 110 is a service platform for providing a baseline model, and may provide the baseline model to the at least one edge server 120. The edge server 120 is a server that has a demand for a baseline model, that is, the edge server 120 needs to obtain the baseline model from the central server 110 and then implement artificial intelligence processing based on the baseline model.

[0068] The first relay device 130 is a network device (e.g., a router, a switch, etc.) connected with the central server 110, and is configured to forward the baseline model transmitted by the central server 110 to the edge server 120, and forward the baseline model transmitted by the edge server 120 to the central server 110. The second relay device 140 is a network device connected with the edge server 120, and is configured to forward the baseline model transmitted by the central server 110 to the edge server 120, and forward the baseline model transmitted by the edge server 120 to the central server 110.

[0069] Embodiments of the present application provide a data processing method. As shown in FIG. 2, the method includes step 201 to step 207.

[0070] At step 201, a central server transmits an initial baseline model to an edge server.

[0071] Exemplarily, the central server may acquire a large amount of sample data without limitation on how to acquire the sample data. For each piece of sample data, the sample data includes label information, e.g., actual category and/or target box, etc., which is not limited. For example, for an application scenario of license plate recognition, the sample data may be a license plate image, the target box may include coordinate information of a rectangular box in the license plate image (e.g., coordinates of an upper left corner of the rectangular box, width and height of the rectangular box, etc.), and the actual category represents a license plate identifier in an area of the rectangular box.

[0072] The central server trains a neural network (e.g., a convolutional neural network) with the sample data and the label information of the sample data to obtain a baseline model, which is referred to as an initial baseline model for convenience of distinction. For example, a large amount of sample data and the label information corresponding to the sample data are input to the neural network, so as to train various parameters in the neural network with these sample data and label information. After the training of the neural network is completed, the trained neural network may be the initial baseline model.

[0073] During the training of the neural network, output data corresponding to the sample data may be obtained after the sample data is input to the neural network. The output data is a feature vector, based on which it may be determined whether the neural network has converged. For example, a loss value of a loss function (which may be configured empirically) is determined from the feature vector, and it is determined, from the loss value, whether the neural network has converged. For example, if the loss value is less than a preset threshold, the neural network has converged; otherwise, the neural network has not converged.

[0074] If the neural network has converged, the training process is completed and the initial baseline model is obtained. If the neural network has not converged, adjustment of various parameters in the neural network is continued to obtain the adjusted neural network, and then, the sample data and the label information corresponding to the sample data may be input to the adjusted neural network to continue the training of the adjusted neural network, and so on, until the neural network has converged.

[0075] In practical applications, it may be determined whether the neural network has converged in other ways, which are not limited. For example, if the number of iterations reaches a preset number threshold, it is determined that the neural network has converged. For another example, if duration of iterations reaches a preset duration threshold, it is determined that the neural network has converged.

[0076] The central server, after obtaining the initial baseline model, may transmit the initial baseline model to the edge server. In the case of a plurality of edge servers, the central server may transmit the initial baseline model to all or some of the edge servers. The transmitting process will be described below in conjunction with one edge server.

[0077] In the first mode, the central server transmits the initial baseline model to the first relay device, the first relay device transmits the initial baseline model to the second relay device, and the second relay device may transmit the initial baseline model to the edge server, such that the initial baseline model is successfully transmitted to the edge server.

[0078] In the second mode, the central server transmits the initial baseline model to the first relay device, and the first relay device performs a first conversion operation on the initial baseline model to obtain a first low-dimensional baseline model, and transmits the first low-dimensional baseline model to the second relay device. The second relay device performs a second conversion operation on the first low-dimensional baseline model to obtain the initial baseline model, and transmits the initial baseline model to the edge server.

[0079] Exemplarily, the first conversion operation may be an encryption operation, and the second conversion operation may be a decryption operation. Alternatively, the first conversion operation may be a compression operation, and the second conversion operation may be a decompression operation. Alternatively, the first conversion operation may be an encryption operation and a compression operation, and the second conversion operation may be a decryption operation and a decompression operation. However, the above are only examples of the first conversion operation and the second conversion operation, which are not limited thereto.

[0080] In a possible embodiment, the first relay device, after receiving the initial baseline model, performs a compression operation on the initial baseline model to convert the initial baseline model into a first low-dimensional baseline model, and transmits the first low-dimensional baseline model to the second relay device. The second relay device, after receiving the first low-dimensional baseline model, performs a decompression operation on the first low-dimensional baseline model to obtain the decompressed initial baseline model.

[0081] Since a compressed first low-dimensional baseline model is transmitted in the network instead of an uncompressed initial baseline model, the amount of data transmitted can be reduced and the time to transmit the model can be saved.

[0082] In a possible embodiment, the first relay device, after receiving the initial baseline model, performs an encryption operation on the initial baseline model to convert the initial baseline model into a first low-dimensional baseline model, and transmits the first low-dimensional baseline model to the second relay device. The second relay device, after receiving the first low-dimensional baseline model, performs a decryption operation on the first low-dimensional baseline model to obtain the decrypted initial baseline model.

[0083] Since an encrypted first low-dimensional baseline model is transmitted in the network instead of an unencrypted initial baseline model, the initial baseline model can be secured against attackers illegally intercepting the initial baseline model.

[0084] In a possible embodiment, the first relay device, after receiving the initial baseline model, performs a compression operation on the initial baseline model and performs an encryption operation on the compressed initial baseline model to convert the initial baseline model into a first low-dimensional baseline model, and transmits the first low-dimensional baseline model to the second relay device. The second relay device, after receiving the first low-dimensional baseline model, performs a decryption operation on the first low-

dimensional baseline model and performs a decompression operation on the decrypted first low-dimensional baseline model to obtain the decompressed initial baseline model.

[0085] Since a compressed and encrypted first low-dimensional baseline model is transmitted in the network, the amount of data transmitted can be reduced, the time to transmit the model can be saved, and attackers can be prevented from illegally intercepting the initial baseline model.

[0086] In a possible embodiment, the first relay device, after receiving the initial baseline model, performs an encryption operation on the initial baseline model and performs a compression operation on the encrypted initial baseline model to convert the initial baseline model into a first low-dimensional baseline model, and transmits the first low-dimensional baseline model to the second relay device. The second relay device, after receiving the first low-dimensional baseline model, performs a decompression operation on the first low-dimensional baseline model and performs a decryption operation on the decompressed first low-dimensional baseline model to obtain the decrypted initial baseline model.

[0087] In the above embodiments, since the first low-dimensional baseline model has been encrypted and/or compressed from the initial baseline model compared to the initial baseline model, the initial baseline model may be understood as a high-dimensional parameter vector, while the first low-dimensional baseline model may be understood as a low-dimensional parameter vector with a smaller number of parameters.

[0088] In the above embodiments, the initial baseline model may be compressed by using a sparse algorithm, and the first low-dimensional baseline model may be decompressed by using the sparse algorithm. However, the sparse algorithm is just an example of a compression algorithm, which is not limited, and any algorithm capable of compressing the initial baseline model may be used.

[0089] In the above embodiments, the initial baseline model may be encrypted by using a cryptographic algorithm, and the first low-dimensional baseline model may be decrypted by using the cryptographic algorithm. The cryptographic algorithm may be any type of cryptographic algorithm, and there is no limitation on this cryptographic algorithm, as long as it can encrypt the initial baseline model. For example, the cryptographic algorithm may be a domestic commercial cryptographic algorithm such as SM1 (SM is short for Security Manager), SM2, SM3, and SM4, or may be an international commercial cryptographic algorithm such as DES (Data Encryption Standard), AES (Advanced Encryption Standard), and IDEA (International Data Encryption Algorithm).

[0090] At step 202, the edge server trains the initial baseline model with scenario data from the edge server to obtain a target baseline model. That is, the trained initial baseline model is the target baseline model.

[0091] Exemplarily, the edge server may acquire a large amount of scenario data without limitation on how to acquire the scenario data, and the scenario data may be scenario data of the environment where the edge server is located. For each piece of scenario data, the scenario data includes label information, e.g., actual category and/or target box, etc., which is not limited.

[0092] For example, for an application scenario of license plate recognition, the scenario data may be a license plate image, the target box may include coordinate information of a rectangular box in the license plate image (e.g., coordinates of an upper left corner of the rectangular box, width and height of the rectangular box, etc.), and the actual category may represent a license plate identifier in an area of the rectangular box.

[0093] The edge server trains the initial baseline model with the scenario data and the label information of the scenario data to obtain the trained baseline model, which is referred to as a target baseline model for convenience of distinction. For example, a large amount of scenario data and the label information corresponding to the scenario data are input to the initial baseline model, so as to train various neural network parameters in the initial baseline model with these scenario data and label information. After the training of the initial baseline model is completed, the trained initial baseline model may be the target baseline model.

[0094] During the training of the initial baseline model, output data corresponding to the scenario data may be obtained after the scenario data is input to the initial baseline model. The output data is a feature vector, based on which it is determined whether the initial baseline model has converged. For example, a loss value of a loss function (which may be configured empirically) is determined from the feature vector, and it is determined, from the loss value, whether the initial baseline model has converged. For example, if the loss value is less than a preset threshold, the initial baseline model has converged; otherwise, the initial baseline model has not converged.

[0095] If the initial baseline model has converged, the training process is completed and the target baseline model is obtained. If the initial baseline model has not converged, adjustment of various neural network parameters in the initial baseline model is continued to obtain the adjusted initial baseline model. The scenario data and the label information are

input to the adjusted initial baseline model to continue the training of the adjusted initial baseline model, and so on, until the initial baseline model has converged.

[0096] In practical applications, it may be determined whether the initial baseline model has converged in other ways, which are not limited. For example, if the number of iterations reaches a preset number threshold, it is determined that the initial baseline model has converged. For another example, if duration of iterations reaches a preset duration threshold, it is determined that the initial baseline model has converged.

[0097] In the above embodiments, the initial baseline model may be trained with the scenario data from the edge server to obtain the target baseline model. Since the scenario data is the data of the environment where the edge server is located, and the scenario data is not transmitted to the central server, that is, the central server does not train the initial baseline model with these scenario data, the edge server trains the initial baseline model with these scenario data such that the target baseline model has sample data information and new knowledge of scenario data, which improves the model performance.

[0098] At step 203, the edge server determines whether to deploy the target baseline model or not. Exemplarily, if the target baseline model is deployed, step 204 is performed; and if the target baseline model is not deployed, step 205 is performed.

[0099] Exemplarily, if the target baseline model has met the performance requirements, the target baseline model may be deployed; and if the target baseline model does not meet the performance requirements, the target baseline model may not be deployed.

[0100] For example, a set of test data is acquired, the set of test data includes a plurality of pieces of test data, and each piece of test data has an actual category corresponding to that test data. For each piece of test data in the set of test data, the test data is input to the target baseline model, and an artificial intelligence processing is performed on the test data through the target baseline model to obtain an artificial intelligence processing result, which is a predicted category of the test data. If the predicted category of the test data is consistent with the actual category of the test data, it means that the recognition result for the test data by the target baseline model is correct; and if the predicted category of the test data is inconsistent with the actual category of the test data, it means that the recognition result for the test data by the target baseline model is incorrect.

[0101] After the above processing on each piece of test data, the number of correct recognition results (denoted as a1) and the number of incorrect recognition results (denoted as a2) may be obtained, and the performance metric of the target baseline model may be

determined from the number of correct recognition results a1 and the number of incorrect recognition results a2. For example, the performance metric may be denoted as a1/(a1+a2). It is apparent that the larger the performance metric is, the better the performance of the target baseline model is.

[0102] As above, if the performance metric of the target baseline model is greater than a preset threshold, it means that the target baseline model has met the performance requirements and the target baseline model may be deployed. If the performance metric of the target baseline model is not greater than the preset threshold, it means that the target baseline model does not meet the performance requirements and the target baseline model may not be deployed.

[0103] However, the above is just an example of determining whether to deploy the target baseline model or not, which is not limited thereto. For example, if the number of iterations (each time the target baseline model is obtained by the edge server through training, it means one iteration, that is, the number of iterations is increased by 1) reaches the preset number threshold, the target baseline model is determined to be deployed. For another example, in response to receiving a user instruction, it is determined whether to deploy the target baseline model based on the user instruction.

[0104] At step 204, the edge server processes application data with the target baseline model.

[0105] Exemplarily, the target baseline model may be deployed on the edge server. The edge server, after acquiring the application data, may input the application data to the target baseline model, and perform an artificial intelligence processing on the application data through the target baseline model, to obtain an artificial intelligence processing result.

[0106] Exemplarily, the edge server may also deploy the target baseline model on a terminal device that is managed by the edge server, e.g., an analog camera, an IPC (Internet Protocol Camera), etc. The terminal device, after acquiring the application data, may input the application data to the target baseline model, and perform an artificial intelligence processing on the application data through the target baseline model, to obtain an artificial intelligence processing result.

[0107] For example, assuming that the target baseline model is configured to implement license plate recognition, the application data are images involving license plates. After the application data is provided to the target baseline model, the target baseline model performs

the artificial intelligence processing on the application data to obtain the artificial intelligence processing result, which may be a license plate identifier.

[0108] At step 205, the edge server transmits the target baseline model to the central server.

[0109] The edge server, after obtaining the target baseline model, transmits the target baseline model to the central server if it is determined that the target baseline model is not deployed. If it is determined that the target baseline model is deployed, the target baseline model may be transmitted to the central server, or the target baseline model may not be transmitted to the central server.

[0110] For the process in which the edge server transmits the target baseline model to the central server, the following modes may be included.

[0111] In the first mode, the edge server transmits the target baseline model to the second relay device, the second relay device transmits the target baseline model to the first relay device, and the first relay device may transmit the target baseline model to the central server, such that the target baseline model is successfully transmitted to the central server.

[0112] In the second mode, the edge server transmits the target baseline model to the second relay device, and the second relay device performs a first conversion operation on the target baseline model to obtain a second low-dimensional baseline model, and transmits the second low-dimensional baseline model to the first relay device. The first relay device performs a second conversion operation on the second low-dimensional baseline model to obtain the target baseline model, and transmits the target baseline model to the central server.

[0113] Exemplarily, the first conversion operation may be an encryption operation, and the second conversion operation may be a decryption operation. Alternatively, the first conversion operation may be a compression operation, and the second conversion operation may be a decompression operation. Alternatively, the first conversion operation may be an encryption operation and a compression operation, and the second conversion operation may be a decryption operation and a decompression operation. However, the above are only examples of the first conversion operation and the second conversion operation, which are not limited thereto.

[0114] In a possible embodiment, the second relay device performs a compression operation on the target baseline model to convert the target baseline model into a second low-dimensional baseline model, and transmits the second low-dimensional baseline model to the first relay device. The first relay device performs a decompression operation on the second low-dimensional baseline model to obtain the target baseline model. Alternatively,

the second relay device performs an encryption operation on the target baseline model to convert the target baseline model into a second low-dimensional baseline model, and transmits the second low-dimensional baseline model to the first relay device. The first relay device performs a decryption operation on the second low-dimensional baseline model to obtain the target baseline model. Alternatively, the second relay device performs a compression operation and an encryption operation on the target baseline model to convert the target baseline model into a second low-dimensional baseline model, and transmits the second low-dimensional baseline model to the first relay device. The first relay device performs a decryption operation and a decompression operation on the second low-dimensional baseline model to obtain the target baseline model.

[0115] At step 206, the central server generates a fusion baseline model based on the target baseline model and the initial baseline model.

[0116] Exemplarily, the central server may perform a fusion operation on the target baseline model of the at least one edge server and the initial baseline model to obtain the fusion baseline model. The fusion operation may include, but is not limited to, one of a weighting operation, an averaging operation, an operation of taking the maximum value, and an operation of taking the minimum value.

[0117] Exemplarily, the target baseline model and the initial baseline model have the same network structure. For example, both the target baseline model and the initial baseline model include network layer 1 and network layer 2, with the network layer 1 including parameters A and B and the network layer 2 including parameters C and D.

[0118] It is assumed that the central server obtains a target baseline model 1 and a target baseline model 2. In the initial baseline model, the value of the parameter A is a11, the value of the parameter B is b11, the value of the parameter C is c11, and the value of the parameter D is d11. In the target baseline model 1, the value of the parameter A is a21, the value of the parameter B is b21, the value of the parameter C is c21, and the value of the parameter D is d21. In the target baseline model 2, the value of the parameter A is a31, the value of the parameter B is b31, the value of the parameter C is c31, and the value of the parameter D is d31.

[0119] On this basis, the central server may perform a fusion operation on the initial baseline model, the target baseline model 1, and the target baseline model 2 to obtain a fusion baseline model. The fusion baseline model may include the network layer 1 and the

network layer 2, with the network layer 1 including parameters A and B and the network layer 2 including parameters C and D.

[0120] In the fusion baseline model, the value of the parameter A is obtained by performing a fusion operation on a11, a21, and a31. For example, the value of the parameter A is the average value of a11, a21, and a31; or, the value of the parameter A is the maximum value among a11, a21, and a31; or, the value of the parameter A is the minimum value among a11, a21, and a31; or, the value of the parameter A is the weighted value of a11, a21, and a31, for example, if the weight of the initial baseline model (e.g., 2) is greater than the weight of the target baseline model (e.g., 1), the value of the parameter A is (a11*2+ a21*1+ a31*1)/4, and if the weight of the initial baseline model (e.g., 1) is less than the weight of the target baseline model (e.g., 2), the value of the parameter A is (a11*1+ a21*2+ a31*2)/5. However, the above are just a few examples, which are not limited thereto.

[0121] In the fusion baseline model, the value of the parameter B is obtained by performing a fusion operation on b11, b21, and b31. For example, the value of the parameter B is the average value of b11, b21, and b31; or, the value of the parameter B is the maximum value among b11, b21, and b31; or, the value of the parameter B is the minimum value among b11, b21, and b31; or, the value of the parameter B is the weighted value of b11, b21, and b31, which is not limited.

[0122] In the fusion baseline model, the value of the parameter C is obtained by performing a fusion operation on c11, c21, and c31. For example, the value of the parameter C is the average value of c11, c21, and c31; or, the value of the parameter C is the maximum value among c11, c21, and c31; or, the value of the parameter C is the minimum value among c11, c21, and c31; or, the value of the parameter C is the weighted value of c11, c21, and c31, which is not limited.

[0123] In the fusion baseline model, the value of the parameter D is obtained by performing a fusion operation on d11, d21, and d31. For example, the value of the parameter D is the average value of d11, d21, and d31; or, the value of the parameter D is the maximum value among d11, d21, and d31; or, the value of the parameter D is the minimum value among d11, d21, and d31; or, the value of the parameter D is the weighted value of d11, d21, and d31, which is not limited.

[0124] At step 207, the central server trains the fusion baseline model to obtain a trained baseline model, determines the trained baseline model as an initial baseline model, and returns to step 201.

[0125] Exemplarily, the central server, after obtaining the fusion baseline model, may train the fusion baseline model with the sample data and the label information of the sample data to obtain the trained baseline model, which can be found in step 201 and will not be repeated herein. The trained baseline model is determined as the initial baseline model, and the method returns to step 201 in which the initial baseline model is transmitted to the edge server.

[0126] As above, in the embodiments of the present application, an edge server may acquire an initial baseline model from a central server, train the initial baseline model with scenario data to obtain a target baseline model, and transmit the target baseline model to the central server, and the central server generates a fusion baseline model based on the target baseline model and the initial baseline model, trains the fusion baseline model, determines the trained baseline model as an initial baseline model, and transmits the initial baseline model to the edge server, and so on. Since the above process may be performed cyclically, the performance of the initial baseline model and the target baseline model is continuously upgraded, and the recognition capability of the initial baseline model and the target baseline model is continuously improved, such that the target baseline model achieves the expected performance and high-accuracy recognition capability. After several iterations, the edge server may obtain the target baseline model with better performance, the target baseline model can match the environment where the edge server is located, and the intelligent analysis results have higher accuracy. In the above process, the edge server transmits the target baseline model to the central server instead of the scenario data (e.g., license plate images), so as to protect the privacy of the scenario data and not transmit the scenario data to the central server. Since the target baseline model is trained with the scenario data, the information of the scenario data can be reflected in the initial baseline model and the target baseline model.

[0127] In a possible embodiment, in the field of intelligent transportation, the scenario data from the edge server may be license plate images, the sample data of the central server may be license plate images, and the target baseline model is configured to perform license plate recognition processing on the application data, to recognize license plate identifiers in the license plate images. In this application scenario, a privacy-removal license plate recognition method is proposed in this embodiment, which can achieve good license plate recognition capability without involving privacy data of users, that is, without transmitting the private license plate images to the central server.

[0128] The baseline models for the edge server and the central server have the same network structure. The initial baseline model trained by the central server has the same network structure as the target baseline model trained by the edge server. As shown in FIG. 3, a schematic diagram of the network structure of the baseline model is illustrated, and both the initial baseline model and the target baseline model have the same network structure.

[0129] Since the edge server is not directly connected with the central server, a problem with a single edge server may not directly affect the operation of the central server, and the edge server may be easily added or removed.

[0130] Exemplarily, the edge server may also be referred to as an end-side server, where the edge server is maintained by a legal (satisfying local regulations) regional support provider/agent, and the edge server has a model training capability. The central server is maintained by a recognition system service provider, and the central server has a model training capability. The edge server may obtain the initial baseline model from the central server to obtain a better initial recognition capability, and train the initial baseline model with local license plate images to obtain the target baseline model, such that the target baseline model can be adapted to the edge server. The central server may obtain the target baseline model transmitted by the edge server, and complete the fusion of the target baseline model and the initial baseline model, such that the initial baseline model can be further upgraded with stronger generalization capability.

[0131] In the above application scenario, embodiments of the present application provide a data processing method, including steps S1 to S12.

[0132] At step S1, an initial baseline model is obtained by a central server through training with sample data.

[0133] At step S2, the central server transmits the initial baseline model to a first relay device.

[0134] At step S3, the first relay device performs a first conversion operation on the initial baseline model to obtain a first low-dimensional baseline mode, and transmits the first low-dimensional baseline model to a second relay device.

[0135] Exemplarily, in the case of a plurality of second relay devices, the first low-dimensional baseline model may be transmitted to each second relay device. Hereinafter, one second relay device will be used as an example for description.

[0136] At step S4, the second relay device performs a second conversion operation on the first low-dimensional baseline model to obtain the initial baseline model, and transmits the initial baseline model to the edge server.

[0137] Exemplarily, in the case of a plurality of second relay devices, for each second relay device, the initial baseline model may be transmitted to the edge server connected with this second relay device, i.e., the initial baseline model may be transmitted to a plurality of edge servers. For ease of description, the processing by one edge server will be described as an example below.

[0138] At step S5, the edge server trains the initial baseline model with scenario data from the edge server to obtain a target baseline model, that is, the trained initial baseline model is the target baseline model.

[0139] At step S6, the edge server determines whether to deploy the target baseline model or not. Exemplarily, step S7 is performed if the target baseline model is deployed, and step S8 is performed if the target baseline model is not deployed.

[0140] At step S7, the edge server processes application data with the target baseline model.

[0141] At step S8, the edge server transmits the target baseline model to the second relay device.

[0142] At step S9, the second relay device performs a first conversion operation on the target baseline model to obtain a second low-dimensional baseline model, and transmits the second low-dimensional baseline model to the first relay device.

[0143] At step S10, the first relay device performs a second conversion operation on the second low-dimensional baseline model to obtain the target baseline model, and transmits the target baseline model to the central server.

[0144] At step S11, the central server generates a fusion baseline model based on the target baseline model and the initial baseline model.

[0145] At step S12, the central server trains the fusion baseline model to obtain a trained baseline model, determines the trained baseline model as an initial baseline model, and returns to step S2.

[0146] As above, since the above process may be performed cyclically, the performance of the initial baseline model of the central server can be continuously upgraded, and the license plate recognition capability can be continuously improved, while protecting the data privacy of each edge server. With an increase in the number of interactions between the

central server and the edge server, the performance of the initial baseline model and the target baseline model can be continuously improved to achieve the expected performance.

[0147] Based on the same application concept as the method described above, embodiments of the present application further provide another data processing method, which can be applied to an edge server. As shown in FIG. 4, the method may include steps 401-404.

[0148] At step 401, an initial baseline model is acquired from a central server.

[0149] At step 402, the initial baseline model is trained with scenario data from the edge server to obtain a target baseline model, and it is determined whether to deploy the target baseline model or not. Exemplarily, step 403 is performed if the target baseline model is deployed, and step 404 is performed if the target baseline model is not deployed.

[0150] At step 403, application data is processed with the target baseline model.

[0151] At step 404, the target baseline model is transmitted to the central server, such that the central server generates a fusion baseline model based on the target baseline model and the initial baseline model, and reacquires an initial baseline model based on the fusion baseline model, and the method returns to the operation of acquiring the initial baseline model from the central server, that is, returns to step 401. In this way, the edge server may reacquire the initial baseline model from the central server.

[0152] Based on the same application concept as the method described above, embodiments of the present application further provide another data processing method, which can be applied to a central server. As shown in FIG. 5, the method may include steps 501-504.

[0153] At step 501, an initial baseline model is transmitted to an edge server, such that the edge server trains the initial baseline model with scenario data from the edge server to obtain a target baseline model.

[0154] At step 502, the target baseline model is acquired from the edge server.

[0155] At step 503, a fusion baseline model is generated based on the target baseline model and the initial baseline model.

[0156] At step 504, the fusion baseline model is trained, and the trained baseline model is determined as an initial baseline model, and the method returns to the operation of transmitting the initial baseline model to the edge server, that is, returns to step 501.

[0157] Based on the same application concept as the method described above, embodiments of the present application provide a data processing apparatus, which is

applied to an edge server. FIG. 6A is a schematic structural diagram of the data processing apparatus. As shown in FIG. 6A, the apparatus includes:

[0158] an acquiring module 611, configured to acquire an initial baseline model from a central server;

[0159] a training module 612, configured to train the initial baseline model with scenario data from the edge server to obtain a target baseline model, and determine whether to deploy the target baseline model or not; and

[0160] a transmitting module 613, configured to transmit the target baseline model to the central server if the target baseline model is not deployed, such that the central server generates a fusion baseline model based on the target baseline model and the initial baseline model, and reacquires an initial baseline model based on the fusion baseline model.

[0161] Exemplarily, the data processing apparatus may further include a processing module, configured to process application data with the target baseline model if the target baseline model is deployed.

[0162] Based on the same application concept as the method described above, embodiments of the present application provide a data processing apparatus, which is applied to a central server. FIG. 6B is a schematic structural diagram of the data processing apparatus. As shown in FIG. 6B, the apparatus includes:

[0163] a transmitting module 621, configured to transmit an initial baseline model to an edge server, such that the edge server trains the initial baseline model with scenario data from the edge server to obtain a target baseline model;

[0164] an acquiring module 622, configured to acquire the target baseline model from the edge server; and

[0165] a generating module 623, configured to generate a fusion baseline model based on the target baseline model and the initial baseline model, train the fusion baseline model, and determine the trained baseline model as an initial baseline model.

[0166] The generating module 623, when generating the fusion baseline model based on the target baseline model and the initial baseline model, is further configured to: perform a fusion operation on the target baseline model of at least one edge server and the initial baseline model to obtain the fusion baseline model, where the fusion operation includes one of a weighting operation, an averaging operation, an operation of taking the maximum value, and an operation of taking the minimum value.

[0167] Based on the same application concept as the method described above, embodiments of the present application provide an edge server. As shown in FIG. 7A, the edge server includes a processor 711 and a machine-readable storage medium 712 storing machine-executable instructions executable by the processor 711, where the processor 711 is configured to execute the machine-executable instructions to implement operations including:

[0168] acquiring an initial baseline model from a central server;

[0169] training the initial baseline model with scenario data from the edge server to obtain a target baseline model, and determining whether to deploy the target baseline model or not; and

[0170] transmitting the target baseline model to the central server if the target baseline model is not deployed, such that the central server generates a fusion baseline model based on the target baseline model and the initial baseline model, and reacquires an initial baseline model based on the fusion baseline model, and returning to the operation of acquiring the initial baseline model from the central server.

[0171] Based on the same application concept as the method described above, embodiments of the present application provide a central server. As shown in FIG. 7B, the central server includes a processor 721 and a machine-readable storage medium 722 storing machine-executable instructions executable by the processor 721, where the processor 721 is configured to execute the machine-executable instructions to implement operations including:

[0172] transmitting an initial baseline model to an edge server, such that the edge server trains the initial baseline model with scenario data from the edge server to obtain a target baseline model;

[0173] acquiring the target baseline model from the edge server;

[0174] generating a fusion baseline model based on the target baseline model and the initial baseline model; and

[0175] training the fusion baseline model, determining the trained baseline model as an initial baseline model, and returning to the operation of transmitting the initial baseline model to the edge server.

[0176] Based on the same application concept as the methods described above, embodiments of the present application further provide a machine-readable storage medium having a number of computer instructions stored therein, where the computer instructions,

when executed by a processor, can implement the data processing methods disclosed in the above examples of the present application.

[0177] The machine-readable storage medium described above may be any electronic, magnetic, optical, or other physical storage device that may contain or store information such as executable instructions, data, and the like. For example, the machine-readable storage medium may include RAM (Radom Access Memory), volatile memory, non-volatile memory, flash memory, storage drive (such as hard disk drive), solid-state drive, any type of storage disk (such as compact disk, DVD, etc.), or similar storage medium, or a combination thereof.

[0178] The systems, apparatuses, modules, or units illustrated in the above embodiments may be specifically implemented by computer chips or entities, or by products with certain functions. An exemplary device for implementation is a computer, which may be in the form of a personal computer, laptop computer, cellular phone, camera phone, smart phone, personal digital assistant, media player, navigation device, email sending and receiving device, game console, tablet computer, wearable device, or a combination of any of these devices.

[0179] For the convenience of description, when describing the above apparatuses, the apparatuses are divided into various units by functions to be described separately. However, functions of various units can be implemented in the same one or more software and/or hardware when implementing the present application.

[0180] Those skilled in the art should understand that embodiments of the present application may be provided as methods, systems, or computer program products. Accordingly, the present application may take the form of an entirely hardware embodiment, an entirely software embodiment, or an embodiment combining software and hardware aspects. Furthermore, the embodiments of the present application may take the form of a computer program product embodied in one or more computer-usable storage media (including, but not limited to, disk storage, CD-ROM, optical storage, etc.) containing computer-usable program codes therein.

[0181] The present application is described with reference to flowcharts and/or block diagrams of methods, devices (systems), and computer program products according to embodiments of the present application. It should be understood that each of flows and/or blocks in the flowchart and/or block diagram, and a combination of flows and/or blocks in the flowchart and/or block diagram may be implemented by computer program instructions.

These computer program instructions may be provided to a processor of a general-purpose computer, a special-purpose computer, an embedded processor, or other programmable data processing device to produce a machine such that the instructions executed by the processor of the computer or other programmable data processing device produce an apparatus for implementing functions specified in one or more flows in the flowchart and/or one or more blocks in the block diagram.

[0182] Moreover, these computer program instructions may also be stored in a computer-readable memory capable of directing a computer or other programmable data processing device to operate in a particular manner, such that the instructions stored in the computer-readable memory produce a manufactured product including an instruction apparatus that implements the functions specified in one or more flows in the flowchart and/or one or more blocks in the block diagram.

[0183] These computer program instructions may also be loaded into a computer or other programmable data processing device, causing a series of operational steps to be performed in the computer or other programmable data processing device to produce a computer-implemented processing, such that the instructions executed in the computer or other programmable data processing device provide steps for implementing the functions specified in one or more flows in the flowchart and/or one or more blocks in the block diagram.

[0184] The foregoing are only embodiments of the present application, and are not intended to limit the present application. For those skilled in the art, the present application is subject to various modifications and changes. Any modification, equivalent substitution, improvement, etc. made within the spirit and principle of the present application shall be included in the scope of the claims of the present application.

CLAIMS

1. A data processing method, comprising:

transmitting, by a central server, an initial baseline model to an edge server;

training, by the edge server, the initial baseline model with scenario data from the edge server to obtain a target baseline model, and determining whether to deploy the target baseline model or not;

transmitting, by the edge server, the target baseline model to the central server if it is determined that the target baseline model is not deployed; and

generating, by the central server, a fusion baseline model based on the target baseline model and the initial baseline model, training the fusion baseline model, determining the trained baseline model as an initial baseline model, and returning to an operation of transmitting, by the central server, the initial baseline model to the edge server.

2. The method according to claim 1, wherein transmitting, by the central server, the initial baseline model to the edge server, comprises:

transmitting, by the central server, the initial baseline model to a first relay device;

performing, by the first relay device, a first conversion operation on the initial baseline model to obtain a first low-dimensional baseline mode, and transmitting the first low-dimensional baseline model to a second relay device; and

performing, by the second relay device, a second conversion operation on the first low-dimensional baseline model to obtain the initial baseline model, and transmitting the initial baseline model to the edge server.

3. The method according to claim 1, wherein transmitting, by the edge server, the target baseline model to the central server, comprises:

transmitting, by the edge server, the target baseline model to a second relay device;

performing, by the second relay device, a first conversion operation on the target baseline model to obtain a second low-dimensional baseline model, and transmitting the second low-dimensional baseline model to a first relay device; and

performing, by the first relay device, a second conversion operation on the second low-dimensional baseline model to obtain the target baseline model, and transmitting the target baseline model to the central server.

4. The method according to claim 2 or 3, wherein

the first conversion operation is an encryption operation, and the second conversion operation is a decryption operation; or

the first conversion operation is a compression operation, and the second conversion operation is a decompression operation; or

the first conversion operation is an encryption operation and a compression operation, and the second conversion operation is a decryption operation and a decompression operation.

5. The method according to claim 1, wherein the edge server comprises at least one edge server, and generating, by the central server, the fusion baseline model based on the target baseline model and the initial baseline model, comprises:

performing, by the central server, a fusion operation on the target baseline model of the at least one edge server and the initial baseline model to obtain the fusion baseline model,

wherein the fusion operation comprises one of a weighting operation, an averaging operation, an operation of taking a maximum value, and an operation of taking a minimum value.

6. The method according to claim 1, wherein after determining whether to deploy the target baseline model or not, the method further comprises:

processing, by the edge server, application data with the target baseline model if it is determined that the target baseline model is deployed.

7. A data processing method, applied to an edge server, the method comprising:

acquiring an initial baseline model from a central server;

training the initial baseline model with scenario data from the edge server to obtain a target baseline model, and determining whether to deploy the target baseline model or not; and

if it is determined that the target baseline model is not deployed, transmitting the target baseline model to the central server, such that the central server generates a fusion baseline model based on the target baseline model and the initial baseline model, and reacquires an initial baseline model based on the fusion baseline model, and returning to an operation of acquiring the initial baseline model from the central server.

8. A data processing method, applied to a central server, the method comprising:

transmitting an initial baseline model to an edge server, such that the edge server trains the initial baseline model with scenario data from the edge server to obtain a target baseline model;

acquiring the target baseline model from the edge server;

generating a fusion baseline model based on the target baseline model and the initial baseline model; and

training the fusion baseline model, determining the trained baseline model as an initial baseline model, and returning to an operation of transmitting the initial baseline model to the edge server.

9. A data processing apparatus, applied to an edge server, the apparatus comprising:

an acquiring module, configured to acquire an initial baseline model from a central server;

a training module, configured to train the initial baseline model with scenario data from the edge server to obtain a target baseline model, and determine whether to deploy the target baseline model or not; and

a transmitting module, configured to transmit the target baseline model to the central server if it is determined that the target baseline model is not deployed, such that the central server generates a fusion baseline model based on the target baseline model and the initial baseline model, and reacquires an initial baseline model based on the fusion baseline model.

10. A data processing apparatus, applied to a central server, the apparatus comprising:

a transmitting module, configured to transmit an initial baseline model to an edge server, such that the edge server trains the initial baseline model with scenario data from the edge server to obtain a target baseline model;

an acquiring module, configured to acquire the target baseline model from the edge server; and

a generating module, configured to generate a fusion baseline model based on the target baseline model and the initial baseline model, train the fusion baseline model, and determine the trained baseline model as an initial baseline model.

11. An edge server, comprising a processor and a machine-readable storage medium storing machine-executable instructions executable by the processor,

wherein the processor is configured to execute the machine-executable instructions to implement operations comprising:

acquiring an initial baseline model from a central server;

training the initial baseline model with scenario data from the edge server to obtain a target baseline model, and determining whether to deploy the target baseline model or not; and

transmitting the target baseline model to the central server if it is determined that the target baseline model is not deployed, such that the central server generates a fusion baseline model based on the target baseline model and the initial baseline model, and reacquires an initial baseline model based on the fusion baseline model, and returning to the operation of acquiring the initial baseline model from the central server.

12. A central server, comprising a processor and a machine-readable storage medium storing machine-executable instructions executable by the processor,

wherein the processor is configured to execute the machine-executable instructions to implement operations comprising:

transmitting an initial baseline model to an edge server, such that the edge server trains the initial baseline model with scenario data from the edge server to obtain a target baseline model;

acquiring the target baseline model from the edge server;

generating a fusion baseline model based on the target baseline model and the initial baseline model; and

training the fusion baseline model, determining the trained baseline model as an initial baseline model, and returning to the operation of transmitting the initial baseline model to the edge server.