

METHODS AND APPARATUSES FOR TRANSFORMING NEURAL NETWORK MODEL, AND METHODS AND APPARATUSES FOR POSITIONING MODEL PRECISION

CROSS REFERENCE TO RELATED APPLICATIONS

[0001] This application claims priority to Chinese Patent Application No. 202010129010.0 filed with Chinese Patent Office on February 28, 2020, the entire content of which is incorporated herein by reference.

TECHNICAL FIELD

[0002] The present disclosure relates to the field of artificial intelligence (AI) technologies, and in particular to a method and an apparatus for transforming a neural network model, and a method and an apparatus for positioning a model precision.

BACKGROUND

[0003] Nowadays, a variety of deep learning model frameworks have been developed in the industry and each framework has its own network structure definition and tool kit. Because different frameworks have their respective advantages in aspects such as training, deployment, and platform compatibility and the like, it is often required to perform transformation between frameworks for a deep learning model. The neural network model transformation refers to transforming a neural network model defined under one framework to be under another framework.

[0004] However, after the neural network model is transformed, it is required to verify a model precision manually by a programmer, resulting in low precision verification efficiency.

SUMMARY

[0005] The present disclosure provides a technical solution for transforming a neural network model.

[0006] According to a first aspect, provided is a method of transforming a neural network model. The method includes: obtaining, by a model transformation apparatus, a first neural network model under a source framework, model input data and information of a target framework; obtaining, by the model transformation apparatus, a second neural network

model under the target framework by performing model transformation on the first neural network model under the source framework based on the model input data; obtaining, by the model transformation apparatus, a precision verification result of the second neural network model based on a first reasoning result and a second reasoning result obtained by processing the model input data respectively through the first neural network model and the second neural network model.

[0007] The model transformation apparatus can perform model transformation based on a user instruction. The user instruction is used to instruct transforming the first neural network model under the source framework to be under the target framework, and the user instruction includes the first neural network model under the source framework, the model input data and the information of the target framework. The model transformation apparatus can also perform model transformation based on an internal operation instruction, for example, a model transformation instruction. The internal operation instruction is used to instruct transforming the first neural network model under the source framework to be under the target framework. The internal operation instruction includes the first neural network model under the source framework, the model input data and the information of the target framework.

[0008] In one possible implementation, the model transformation apparatus obtains the first reasoning result of the first neural network model for the model input data in the process of performing model transformation on the first neural network model.

[0009] In one possible implementation, the model transformation apparatus locally stores the first reasoning result.

[0010] In one possible implementation, the model transformation apparatus obtains the second reasoning result by processing the model input data through the second neural network model.

[0011] In one possible implementation, the model transformation apparatus obtains a precision verification result of the second neural network model by comparing the first reasoning result and the second reasoning result.

[0012] In one possible implementation, obtaining the second neural network model under the target framework by performing model transformation on the first neural network model under the source framework includes: obtaining the second neural network model under the target framework by performing model transformation on the first neural network model under the source framework through a first process; the method further includes: creating a

second process through the first process; calling the second process through the first process in such a way that the second reasoning result is obtained by processing the model input data through the second neural network model in the second process.

[0013] In one possible implementation, obtaining, by the model transformation apparatus, the precision verification result of the second neural network model based on the first reasoning result and the second reasoning result obtained by processing the model input data respectively through the first neural network model and the second neural network model includes: if a difference between the first reasoning result and the second reasoning result is within a preset error range, determining the precision verification result of the second neural network model as precision-aligned, or, if the difference between the first reasoning result and the second reasoning result is outside the preset error range, determining the precision verification result of the second neural network model as precision-unaligned.

[0014] In one possible implementation, the method further includes: in response to determining the precision verification result of the second neural network model as precision-unaligned, positioning, by the model transformation apparatus, a precision-unaligned target network layer in the second neural network model; outputting, by the model transformation apparatus, information of the target network layer.

[0015] In one possible implementation, positioning the precision-unaligned target network layer in the second neural network model includes: establishing, by the model transformation apparatus, a mapping relationship between network layers in the first neural network model and the second neural network model; determining, by the model transformation apparatus, the precision-unaligned target network layer in the second neural network model based on output data of each network layer obtained by processing the model input data respectively through the first neural network model and the second neural network model and the mapping relationship.

[0016] In one possible implementation, the target network layer is a network layer which is precision-unaligned earliest in a plurality of network layers included in the second neural network model.

[0017] In another possible implementation, target network layers are a plurality of network layers which are precision-unaligned in the second neural network model. For example, the target network layers include both the network layer which is precision-unaligned earliest and the subsequent network layers of the network layer.

[0018] In one possible implementation, the information of the target network layer includes at least one of: an identifier of the target network layer, a parameter of the target network layer, output data of the target network layer, and an error between the output data and output data of a network layer corresponding to the target network layer in the first neural network model.

[0019] In one possible implementation, the model transformation apparatus re-performs model transformation on the first neural network model, where in the process of re-performing model transformation, output data of each network layer in the first neural network model is obtained, and/or, the mapping relationship between network layers in the first neural network model and the second neural network model is established.

[0020] In one possible implementation, obtaining, by the model transformation apparatus, the second neural network model under the target framework by performing model transformation on the first neural network model under the source framework includes: establishing, by the model transformation apparatus, a network topology by processing the model input data through the first neural network model, where the network topology includes information of the network layers and a connection relationship of the network layers in the first neural network model; obtaining, by the model transformation apparatus, the second neural network model based on the network topology and the information of the target framework.

[0021] According to a second aspect, provided is an apparatus for transforming a model. The apparatus includes: an obtaining unit, configured to obtain a first neural network model under a source framework, model input data, and information of a target framework; a transforming unit, configured to obtain a second neural network model under the target framework by performing model transformation on the first neural network model under the source framework; a verifying unit, configured to obtain a precision verification result of the second neural network model based on a first reasoning result and a second reasoning result obtained by processing the model input data respectively through the first neural network model and the second neural network model.

[0022] According to a third aspect, provided is an electronic device, including a memory and a processor. The memory stores program instructions, and the processor is configured to invoke the program instructions stored in the memory to implement the method according to the first aspect or any item of the first aspect.

[0023] According to a fourth aspect, provided is a computer readable storage medium, storing instructions. When run on a computer, the instructions cause the computer to implement the method according to the first aspect or any item of the first aspect.

[0024] According to a fifth aspect, provided is a computer program product including instructions. When run on a computer, the instructions cause the computer to implement the method according to the first aspect or any item of the first aspect.

[0025] The technical solution for transforming a neural network model in the present disclosure has the following technical effects.

[0026] According to the reasoning results of the first neural network model under the source framework and the second neural network model under the target framework for the same model input data, the precision verification result of the second neural network model can be automatically obtained, thus improving the precision verification efficiency of the transformed neural network model.

[0027] The embodiments of the present disclosure further provide a solution for positioning a model precision.

[0028] According to a sixth aspect, provided is a method of positioning a model precision. The method includes: obtaining, by a model precision positioning apparatus, a first neural network model under a source framework, model input data and information of a target framework; obtaining, by the model precision positioning apparatus, a second neural network model under the target framework by performing model transformation on the first neural network model; positioning, by the model precision positioning apparatus, a precision-unaligned target network layer in the second neural network model based on output data of each network layer obtained by processing the model input data respectively through the first neural network model and the second neural network model; outputting, by the model precision positioning apparatus, information of the target network layer.

[0029] In a case that it is known that the second neural network model and the first neural network model are unaligned in precision, the model precision positioning apparatus may perform model precision positioning based on a user instruction. The user instruction is used to instruct positioning the precision-unaligned target network layer in the second neural network model. The user instruction includes the first neural network model under the source framework, the model input data, and the information of the target framework. For example, when obtaining a result that the second neural network model and the first neural network model are unaligned in precision, the user sends the user instruction to the model

precision positioning apparatus. The model precision positioning apparatus may also perform model precision positioning based on an internal operation instruction, for example, a model precision positioning instruction. For example, when obtaining an instruction that the second neural network model and the first neural network model are unaligned in precision, the model precision positioning apparatus triggers the model precision positioning instruction. The internal operation instruction is used to instruct positioning the precision-unaligned target network layer in the second neural network model, and the internal operation instruction includes the first neural network model under the source framework, the model input data, and the information of the target framework. In one possible implementation, the output data of each network layer in the second neural network model is obtained by processing the model input data through the second neural network model.

[0030] In one possible implementation, the output data generated by each network layer of the first neural network model for processing the model input data is obtained during the model transformation.

[0031] In one possible implementation, the target network layer is a network layer which is precision-unaligned earliest in a plurality of network layers included in the second neural network model.

[0032] In another possible implementation, the target network layer is a plurality of network layers which are precision-unaligned in the second neural network model. For example, the target network layer includes both the network layer which is precision-unaligned earliest and the subsequent network layers of the network layer.

[0033] In one possible implementation, the target network layer may be determined by comparing the output data of corresponding network layers in the first neural network model and the second neural network model.

[0034] In one possible implementation, if a difference between the output data of the network layer included in the second neural network model and the output data of the corresponding network layer of the first neural network model is outside a preset error range, the network layer in the second neural network model is determined as a precision-unaligned network layer.

[0035] In one possible implementation, the information of the target network layer includes at least one of: an identifier of the target network layer, a parameter of the target network layer, output data of the target network layer, and an error between the output data

and output data of a network layer corresponding to the target network layer in the first neural network model.

[0036] In one possible implementation, the method further includes: establishing, by the model precision positioning apparatus, a mapping relationship between network layers in the first neural network model and the second neural network model based on the model transformation; positioning, by the model precision positioning apparatus, the precision-unaligned target network layer in the second neural network model based on the output data of each network layer obtained by processing the model input data respectively through the first neural network model and the second neural network model includes: determining, by the model precision positioning apparatus, the precision-unaligned target network layer in the second neural network model based on the output data of each network layer in the first neural network model and the second neural network model and the mapping relationship.

[0037] In one possible implementation, determining, by the model precision positioning apparatus, the precision-unaligned target network layer in the second neural network model based on the output data of each network layer in the first neural network model and the second neural network model and the mapping relationship includes: comparing, by the model precision positioning apparatus, the output data of corresponding network layers in the first neural network model and the second neural network model sequentially based on the mapping relationship; in response to determining that a difference between the output data of corresponding network layers in the first neural network model and the second neural network model is outside a preset error range, determining the network layer in the corresponding network layers which belongs to the second neural network model as the target network layer.

[0038] In one possible implementation, obtaining, by the model precision positioning apparatus, the second neural network model under the target framework by performing model transformation on the first neural network model under the source framework includes: establishing, by the model precision positioning apparatus, a network topology by processing the model input data through the first neural network model, wherein the network topology includes information of the network layers and a connection relationship of the network layers in the first neural network model; obtaining, by the model precision positioning apparatus, the second neural network model based on the network topology and the information of the target framework.

[0039] According to a seventh aspect, provided is an apparatus for positioning a model precision. The apparatus includes: an obtaining unit, configured to obtain a first neural network model under a source framework, model input data and information of a target framework; a transforming unit, configured to obtain a second neural network model under the target framework by performing model transformation on the first neural network model; a positioning unit, configured to position a precision-unaligned target network layer in the second neural network model based on output data of each network layer obtained by processing the model input data respectively through the first neural network model and the second neural network model; an outputting unit, configured to output information of the target network layer.

[0040] According to an eighth aspect, provided is an electronic device, including a memory and a processor. The memory stores program instructions and the processor is configured to invoke the program instructions stored in the memory to implement the method according to the sixth aspect or any item of the sixth aspect.

[0041] According to a ninth aspect, provided is a computer readable storage medium storing instructions. When run on a computer, the instructions cause the computer to implement the method according to the sixth aspect or any item of the sixth aspect.

[0042] According to a tenth aspect, provided is a computer program product including instructions. When run on a computer, the instructions cause the computer to implement the method according to the sixth aspect or any item of the sixth aspect.

[0043] The solution for positioning a model precision in the present disclosure has the following technical effects.

[0044] According to the output data of each network layer obtained respectively by processing the same model input data through the first neural network model under the source framework and the second neural network model under the target framework, the precision-unaligned target network layer in the second neural network model can be positioned, thus accurately positioning the precision-unaligned target network layer.

BRIEF DESCRIPTION OF THE DRAWINGS

[0045] FIG. 1 is a flowchart illustrating a method of transforming a neural network model according to an embodiment of the present disclosure.

[0046] FIG. 2 is a schematic diagram illustrating a process of transforming a neural network model according to an embodiment of the present disclosure.

[0047] FIG. 3 is another flowchart illustrating a method of transforming a neural network model according to an embodiment of the present disclosure.

[0048] FIG. 4 is a schematic diagram illustrating a network topology obtained in a process of positioning a target network layer according to an embodiment of the present disclosure.

[0049] FIG. 5 is a flowchart illustrating a method of positioning a model precision according to an embodiment of the present disclosure.

[0050] FIG. 6 is another flowchart illustrating a method of positioning a model precision according to an embodiment of the present disclosure.

[0051] FIG. 7 is a structural schematic diagram illustrating an apparatus for transforming a model according to an embodiment of the present disclosure.

[0052] FIG. 8 is a structural schematic diagram illustrating an apparatus for positioning a model precision according to an embodiment of the present disclosure.

[0053] FIG. 9 is a schematic diagram illustrating a hardware structure of an electronic device according to an embodiment of the present disclosure.

DETAILED DESCRIPTION OF THE EMBODIMENTS

[0054] The technical solutions of embodiments of the present disclosure will be described clearly and fully below in combination with drawings in the embodiments of the present disclosure. It is apparent that the described embodiments are merely some of embodiments of the present disclosure rather than all embodiments. Other embodiments achieved by those of ordinary skill in the art based on the embodiments in the present disclosure without paying creative work shall all fall into the scope of protection of the present disclosure.

[0055] It is understood that the term “and/or” herein is used only to represent three association relationships for description of associated objects, for example, A and/or B may refer to that A exists alone, or both A and B exist, or B exists alone. Furthermore, the character “/” in the present disclosure generally represents that the objects associated back and forth are in an “or” relationship. The terms “first” and “second” etc. in the present disclosure are used to distinguish different objects rather than describe a specific order.

[0056] FIG. 1 is a flowchart illustrating a method of transforming a neural network model according to an embodiment of the present disclosure. The method includes the following steps.

[0057] At step S101, a model transformation apparatus obtains a first neural network model under a source framework, model input data and information of a target framework.

[0058] Because different frameworks have their respective advantages in aspects such as training, deployment, and platform compatibility and the like, transformation between frameworks can be performed on the neural network model. Each framework has its own network structure definition and tool kit. The transformation of the neural network model refers to transforming a neural network model defined under one framework to be under another framework. For example, a neural network model under tensorflow framework is transformed to be a neural network model under pytorch framework, or, a neural network model under a custom framework is transformed to be a neural network model under a more general framework such as pytorch or tensorflow, and vice versa and the like. A framework corresponding to the neural network model before transformation is called “source framework”, and a framework corresponding to the neural network model after transformation is called “target framework”. The specific implementation of the source framework and the target framework is not defined in the present disclosure.

[0059] The model transformation apparatus may perform model transformation based on a user instruction. The user instruction is used to instruct transforming a first neural network model under the source framework to be under the target framework. The user instruction includes the first neural network model under the source framework, the model input data and the information of the target framework. The model transformation apparatus may also perform model transformation based on an internal operation instruction, for example, a model transformation instruction. The internal operation instruction is used to instruct transforming the first neural network model under the source framework to be under the target framework, and the internal operation instruction includes the first neural network model under the source framework, the model input data and the information of the target framework. The first neural network model is a general neural network model and hence may be a neural network model having any function. The first neural network model includes one or more network layers such as convolutional layer, pooling layer, fully-connected layer and normalization layer and the like. The first neural network model under the source framework includes a structure description file and the like of each network layer in the first neural network model under the source framework. The structure description file is used to describe a name, a layer index and the like of the network layer. The model input data refers to model input data for reasoning of the neural network model, which may also be called test data. The model input data may be one picture, or one segment of words or

the like. The information of the target framework includes an identifier, a format and a tool kit and the like of the target framework.

[0060] At step S102, the model transformation apparatus obtains a second neural network model under the target framework by performing model transformation on the first neural network model under the source framework.

[0061] In order to obtain the second neural network model under the target framework, it is required to perform model transformation on the first neural network model under the source framework, that is, transform the first neural network model to be under the target framework. FIG. 2 is a schematic diagram illustrating a process of transforming a neural network model according to an embodiment of the present disclosure. Framework transformation of the neural network model may be performed using a model transformation apparatus (also referred to as neural network model transformation tool). Each network layer of the second neural network model under the target framework is obtained by respectively transforming each network layer of the first neural network model under the source framework to a format of the target framework using the neural network model transformation tool, thereby obtaining the second neural network model under the target framework. One network layer of the first neural network model may be transformed to one or more network layers of the second neural network model, and one network layer of the second neural network model may correspond to one or more network layers of the first neural network model, which is not limited in the embodiments of the present disclosure.

[0062] In some embodiments, during a process of performing the above model transformation on the first neural network model, the model transformation apparatus may run a reasoning process of the first neural network model for the model input data, so as to obtain a first reasoning/inference result, for example, an output 1 shown in FIG. 2. Alternatively, the first reasoning result may be locally stored or recorded. The first reasoning result may be used to verify a precision of the second neural network model. Therefore, during a model transformation, a reasoning result of the first neural network model for the model input data may be obtained without running the reasoning process one more time, thereby improving the efficiency of the precision verification.

[0063] In some other embodiments, the model transformation apparatus may also re-input the model input data into the first neural network model and run the reasoning process of the first neural network model for the model input data to obtain the first reasoning result.

[0064] In some other embodiments, the model transformation apparatus may pre-store the first reasoning result of the first neural network model for the model input data.

[0065] The model transformation apparatus obtains the second neural network model under the target framework by performing model transformation on the first neural network model under the source framework. Specifically, firstly, the model transformation apparatus obtains a network topology of the first neural network model by performing model analysis on the first neural network model. Alternatively, the model analysis may be implemented by the reasoning of the first neural network model for the model input data or the like, and the network topology may also be referred to as a computing diagram which includes various network layers and connection relationship of various network layers in the first neural network model, as shown in a network topological structure shown in FIG. 4. The network topology may not include the output of each network layer, thus helping to improve the model transformation efficiency.

[0066] Next, the model transformation apparatus obtains the second neural network model based on the network topology and the information of the target framework. After the model transformation apparatus obtains the network topology of the first neural network model, the target framework has its own network structure definition and tool kit. The model transformation apparatus may obtain the second neural network model by performing adaptive network construction on the network topology of the first neural network model using the network structure definition and/or the tool kit of the target framework based on the network topology of the first neural network model.

[0067] At step S103, the model transformation apparatus obtains a precision verification result of the second neural network model based on the first reasoning result obtained by processing the model input data with the first neural network model and a second reasoning result obtained by processing the model input data with the second neural network model.

[0068] After the second neural network model under the target framework is obtained, the above model input data may be input into the second neural network model and each network layer of the second neural network model runs a reasoning process one time for the model input data. A reasoning result output by a previous network layer is taken as input data of a next network layer and until a final reasoning result (i.e., the second reasoning result) of the second neural network model is obtained.

[0069] It is understood that the first reasoning result is a reasoning result output by the last layer of the first neural network model under the source framework for processing the model

input data, or a meaningful reasoning result output by a particular layer counted from last of the first neural network model under the source framework for processing the model input data. Correspondingly, the second reasoning result is a reasoning result output by the last layer of the second neural network model under the target framework for processing the model input data or a meaningful reasoning result output by a particular layer counted from the last of the second neural network model under the target framework for processing the model input data, which is not limited in the embodiments of the present disclosure.

[0070] As shown in FIG. 2, the model input data is respectively input into the first neural network model under the source framework and the second neural network model under the target framework to obtain an output 1(i.e. the first reasoning result) and an output 2(i.e. the second reasoning result).

[0071] After the first reasoning result and the second reasoning result are obtained, because the first reasoning result and the second reasoning result are obtained by respectively running a reasoning process using the same model input data in the first neural network model and the second neural network model, the precision verification result of the second neural network model under the target framework may be obtained based on the first reasoning result and the second reasoning result, for example, by comparing the first reasoning result and the second reasoning result. For another example, a matching result may be obtained by performing one-to-one matching on a plurality of parameter values of the second reasoning result and corresponding parameter values of the first reasoning result, thereby obtaining the above precision verification result. In some embodiments, the precision verification result may indicate a comparison result of model precisions of the first neural network model and the second neural network model, where the comparison result may be, for example, precision-alignment or precision-unalignment; alternatively, the precision verification result may indicate whether the model precision of the second neural network model satisfies a desired or pre-determined criterion, for example, whether the precision is up to criterion or not and the like, which is not limited in the embodiments of the present disclosure.

[0072] In some embodiments, the precision verification result may be output, or the information of the second neural network model may be stored, which is not limited in the embodiments of the present disclosure.

[0073] As shown in FIG. 2, precision verification is performed based on the output 1 of the neural network model under the source framework and the output 2 of the neural network model under the target framework.

[0074] In the method of transforming a neural network model according to an embodiment of the present disclosure, based on the reasoning results obtained by processing the same model input data through the first neural network model under the source framework and the second neural network model under the target framework, the precision verification result of the second neural network model may be obtained automatically, thus improving the precision verification efficiency of the transformed neural network model.

[0075] FIG. 3 is another flowchart illustrating a method of transforming a neural network model according to an embodiment of the present disclosure. The method includes the following steps.

[0076] At step S201, a model transformation apparatus obtains a first neural network model under a source framework, model input data and information of a target framework.

[0077] The specific implementation of the step may be referred to step S101 in the embodiment shown in FIG. 1.

[0078] At step S202, a second neural network model under the target framework is obtained by performing model transformation on the first neural network model under the source framework through a first process, and a first reasoning result obtained by processing the model input data through the first neural network model is recorded.

[0079] Part of the implementation of the step may be referred to step S102 in the embodiment of FIG. 1.

[0080] The above model transformation process is carried out in the first process.

[0081] At step S203, a second process is created through the first process.

[0082] In order to avoid resource conflict in a reasoning process, when reasoning with the same model input data is performed in the second neural network model, a new process, i.e. the second process, can be created through the first process. The second process and the first process occupy different resources.

[0083] It is understood that in a multi-process call, the neural network model transformation tool and the first neural network model under the source framework are generally developed based on a platform, and different execution manners may be adopted based on different target frameworks during framework transformation of the neural network model. Therefore, the created second process may be a process of a system

environment, or a process of this platform. In some embodiments, the neural network model transformation tool and the source framework may be developed based on Python platform. When the target framework is Caffe framework, because the Caffe provides a Python interface of Pycaffe, a new process can be created by using a multiprocessing module of the Python to continue using the target framework in the Python environment. In some embodiments, the neural network model transformation tool and the source framework may be developed based on Python platform. When the target framework supports a call of command line, a new process execution command can be created using `os.system` of an `os` module of the Python in the system environment to start the target framework.

[0084] In the created new process, the second neural network model performs reasoning for the model input data. In this way, some system call conflicts when the second neural network model and the first neural network model are operated in a same process may be avoided in the reasoning process, thus preventing system crash. Therefore, reliability of the reasoning process is improved.

[0085] At step S204, the second process is called by the first process, such that the second neural network model processes the model input data in the second process to obtain a second reasoning result.

[0086] The newly-created second process is called by the first process, and the above model input data is input into the second neural network model and then each network layer of the second neural network model runs a reasoning process one time using the model input data. A reasoning result output by a previous network layer is taken as input data of a next network layer and until a final reasoning result (i.e., the second reasoning result) of the second neural network model is obtained.

[0087] At step S205, it is determined whether a difference between the first reasoning result and the second reasoning result is within a preset error range. If yes, step S206 is performed and otherwise, step S207 is performed.

[0088] At step S206, a precision verification result of the second neural network model is determined as precision-aligned.

[0089] At step S207, the precision verification result of the second neural network model is determined as precision-unaligned.

[0090] The first reasoning result and the second reasoning result are obtained by running reasoning process using the same model input data in the first neural network model and the second neural network model respectively. Therefore, after the first reasoning result and the

second reasoning result are obtained, the precision verification result of the second neural network model under the target framework can be obtained based on the first reasoning result and the second reasoning result. Specifically, if the difference between the first reasoning result and the second reasoning result is within the preset error range, the precision verification result of the second neural network model is determined as precision-aligned. If the difference between the first reasoning result and the second reasoning result is outside the preset error range, the precision verification result of the second neural network model is determined as precision-unaligned.

[0091] At step S208, in response to determining the precision verification result of the second neural network model as precision-unaligned, the model transformation apparatus positions a precision-unaligned target network layer in the second neural network model.

[0092] When the precision verification result of the second neural network model is determined as precision-unaligned, it is required to position a network layer which is precision-unaligned with the first neural network model earliest in the second neural network model, that is, position a network layer whose reasoning result is firstly inconsistent. The positioned network layer is referred to as a target network layer. The target network layer is a network layer which is precision-unaligned earliest in a plurality of network layers included in the second neural network model. Alternatively, the target network layer is a first network layer of a plurality of network layers included in the second neural network model, where a difference between output data of the first network layer and output data of corresponding network layer of the first neural network model is outside the preset error range.

[0093] In order to increase the efficiency of the neural network model transformation tool, the reasoning result (also called output data) of any layer is usually not stored during the above precision verification of the neural network model. When the precision verification result is precision-unaligned, the neural network model transformation tool performs one neural network model transformation and reasoning again for use in precision positioning problem. That is, the model transformation apparatus re-performs model transformation on the first neural network model. A data volume output by each layer of the neural network model is relatively large, and thus the output data of each layer of the neural network model is usually not stored in a memory but locally stored. Therefore, in a process of re-performing model transformation, the output data of each network layer of the first neural network

model is locally stored. When a precision of a particular network layer is verified, the output data of the network layer is loaded to the memory from locality.

[0094] When each network layer of the first neural network model under the source framework is transformed to a format of the target framework respectively by using the neural network model transformation tool, the source framework may run a reasoning process one time on the first neural network model by using the model input data, such that information of the network layers and a connection relationship of the network layers in the first neural network model can be obtained. Next, the model transformation apparatus may obtain the second neural network model by performing adaptive configuration for the network topology of the first neural network model using the network structure definition and the tool kit of the target framework based on the network topology of the first neural network model. Furthermore, in a process of re-performing model transformation, a mapping relationship between the network layers in the first neural network model and the second neural network model may also be established. The mapping relationship between the network layers in the first neural network model and the second neural network model may be one to multiple, multiple to one or one to one, and correspondingly the target network layer may be one or more network layers.

[0095] Afterwards, the same model input data is input into the second neural network model, and the output data of each network layer of the second neural network model for processing the model input data is recorded.

[0096] After obtaining the output data of each network layer in the first neural network model and the second neural network model and the mapping relationship between the network layers in the first neural network model and the second neural network model, the model transformation apparatus can determine a target network layer which is precision-unaligned earliest in the second neural network model based on the output data of each network layer in the first neural network model and the second neural network model and the mapping relationship.

[0097] Specifically, in a precision positioning process, to precisely position the target network layer which is precision-unaligned earliest, a sequence number is configured for each network layer of the first neural network model. The sequence numbers may be in an ascending order. Hence, when positioning a precision, each network layer of the first neural network model may be sequentially called based on the sequence numbers of a plurality of network layers in the first neural network model.

[0098] For each called network layer, a reasoning result output by a previous network layer is input to a next network layer for reasoning so as to obtain a reasoning result of each network layer. Furthermore, the reasoning result of each network layer is locally and correspondingly stored according to the sequence number of each network layer. Further, the reasoning result of each network layer is also assigned an identifier. A plurality of inputs may be present on one network layer. Only the reasoning result output by each layer is stored because except for the initial input data, the input of other network layer is the output of a previous layer. Therefore, when the initial input and the output of each layer in the first neural network model are recorded, it means that the inputs and outputs of all layers are recorded. FIG. 4 is a schematic diagram illustrating a network topology obtained in a process of positioning a target network layer according to an embodiment of the present disclosure. The neural network model includes a plurality of network layers, such as convolutional layer, activation function, normalization layer, Eltwise layer and pooling layer and the like. After data is initially input to the first network layer of the neural network model, the output of each network layer is taken as an input of a next network layer. The reasoning result of the neural network model includes initial input data, and input/output data 1 to 10.

[0099] The second neural network model is obtained by transforming each network layer of the first neural network model to be under the target framework separately. The same model input data is input to the second neural network model and then a reasoning result of each layer of the second neural network model is obtained by using the input of each layer and stored locally. The sequence number of each network layer is taken as an index for storage. Of course, the reasoning process under the target framework also needs to be carried out in a new process.

[0100] After the reasoning result of each network layer of the neural network model under the target framework and the source framework is obtained, the reasoning result of each network layer in the first neural network model under the source framework and the reasoning result of the corresponding network layer in the second neural network model under the target framework may be compared sequentially based on the sequence number of each network layer, so as to position a network layer which is precision-unaligned with the first neural network model earliest in the second neural network model, thus obtaining the target network layer.

[0101] At step S209, the model transformation apparatus outputs information of the target network layer.

[0102] After determining the target network layer, the model transformation apparatus may output the information of the target network layer. The information of the target network layer includes an identifier, a parameter, input data and output result and the like of the network layer. An administrator or user may analyze a cause of the precision un-alignment based on the information of the target network layer.

[0103] In the method of transforming a neural network model according to an embodiment of the present disclosure, based on the reasoning results obtained by processing the same model input data through the first neural network model under the source framework and the second neural network model under the target framework, the precision verification result of the second neural network model can be obtained automatically, thus improving the precision verification efficiency of the transformed neural network model. Furthermore, based on the output data of each network layer obtained by processing the same model input data respectively through the first neural network model under the source framework and the second neural network model under the target framework, the precision-unaligned target network layer in the second neural network model can be positioned, thereby precisely positioning the network layer when the precision is not aligned.

[0104] When the neural network model under the source framework is transformed to be under the target framework, the same model input data is input to the neural network models to determine whether a difference between the output of the source framework and the output of the target framework is within an error range. If the difference is not within the error range, it is required to position which network layer of the neural network model results in different reasoning results under the two frameworks. At present, the open-source neural network model transformation tool does not verify whether the output precisions before and after the transformation of the neural network model are consistent, and thus it is required for a user to manually perform running under two frameworks and store results and then compare the results. In a case of precision un-alignment, the user needs to modify the neural network model based on experiences, output the results of some layers of the modified neural network model, and position a network layer resulting in inconsistent reasoning results under the two frameworks. This manner will take much time and the positioning will become very difficult for some large neural network models.

[0105] FIG. 5 is a flowchart illustrating a method of positioning a model precision according to an embodiment of the present disclosure.

[0106] At step S301, a model precision positioning apparatus obtains a first neural network model under a source framework, model input data, and information of a target framework.

[0107] In this embodiment, the model precision positioning apparatus is only different in name from the model transformation apparatus and the neural network model transformation tool in the above embodiments and can implement the same functions. When it is known that a second neural network model and the first neural network model are unaligned in precision, the model precision positioning apparatus may perform model precision positioning based on a user instruction. The user instruction is used to instruct positioning a precision-unaligned target network layer in the second neural network model. The user instruction includes the first neural network model under the source framework, the model input data and the information of the target framework. For example, When obtaining a result that the second neural network model and the first neural network model are unaligned in precision, the user may send the user instruction to the model precision positioning apparatus. The model precision positioning apparatus may perform model precision positioning based on an internal operation instruction, for example, a model precision positioning instruction. For example, when obtaining an instruction that the second neural network model and the first neural network model are unaligned in precision, the model precision positioning apparatus triggers a model precision positioning instruction. The internal operation instruction is used to instruct positioning a precision-unaligned target network layer in the second neural network model. The internal operation instruction includes the first neural network model under the source framework, the model input data and the information of the target framework.

[0108] The specific implementation of the step may be referred to step S101 of the embodiment of FIG. 1 or to step S201 of the embodiment of FIG. 3.

[0109] At step S302, the model precision positioning apparatus obtains a second neural network model under the target framework by performing model transformation on the first neural network model.

[0110] The specific implementation of the step may be referred to step S102 of the embodiment of FIG. 1 or to step S202 of the embodiment of FIG. 3.

[0111] At step S303, the model precision positioning apparatus positions a precision-unaligned target network layer in the second neural network model based on output data of each network layer obtained by processing the model input data respectively through the first neural network model and the second neural network model.

[0112] During a process of performing the above model transformation on the first neural network model, the model precision positioning apparatus runs a reasoning process of the first neural network model for the model input data to obtain output data of each network layer of the first neural network model for processing the model input data. Alternatively, the output data of each network layer of the first neural network model may be locally stored or recorded.

[0113] In some other embodiments, the model precision positioning apparatus may also re-input the model input data into the first neural network model and run the reasoning process of the first neural network model for the model input data to obtain the output data of each network layer of the first neural network model for processing the model input data.

[0114] In some other embodiments, the model precision positioning apparatus may pre-store the output data of each network layer obtained by performing reasoning for the model input data through the first neural network model.

[0115] After the second neural network model under the target framework is obtained, the above model input data may be input into the second neural network model and each network layer of the second neural network model runs a reasoning process one time using the model input data. A reasoning result output by a previous network layer is taken as input data of a next network layer and thus the output data of each network layer in the second neural network model is obtained.

[0116] A first reasoning result and a second reasoning result are obtained by running a reasoning process using the same model input data in the first neural network model and the second neural network model respectively. Therefore, after the first reasoning result and the second reasoning result are obtained, the precision verification result of the second neural network model under the target framework may be obtained based on the first reasoning result and the second reasoning result, for example, by comparing the first reasoning result and the second reasoning result. For another example, a matching result may be obtained by performing one-to-one matching for a plurality of parameter values of the second reasoning result and corresponding parameter values of the first reasoning result, thereby obtaining the above precision verification result. In some embodiments, the precision verification result may indicate a comparison result of model precisions of the first neural network model and the second neural network model, where the comparison result may be, for example, precision-aligned or precision-unaligned; alternatively, the precision verification result may indicate whether the model precision of the second neural network model satisfies a desired

or pre-determined criterion, for example, whether the precision is up to criterion or not and the like, which is not limited in the embodiments of the present disclosure.

[0117] The output result of each network layer of the first neural network model and the output result of each network layer of the second neural network model are obtained by running a reasoning process using the same model input data in the first neural network model and the second neural network model respectively. Therefore, after the output result of each network layer of the first neural network model and the output result of each network layer of the second neural network model are obtained, a precision-unaligned target network layer in the second neural network model may be positioned based on the output result of each network layer of the first neural network model and the output result of each network layer of the second neural network model.

[0118] In one possible implementation, the target network layer is a network layer which is precision-unaligned earliest in a plurality of network layers included in the second neural network model. Specifically, if a difference between the output data of a network layer included in the second neural network model and the output data of the corresponding network layer of the first neural network model is outside a preset error range, the network layer of the second neural network model is determined as a precision-unaligned network layer.

[0119] In another possible implementation, the target network layer is a plurality of network layers which are precision-unaligned in the second neural network model. For example, the target network layer includes both the network layer which is precision-unaligned earliest and the subsequent network layers of the network layer. The number of the subsequent network layers may be preset.

[0120] Specifically, the reasoning result of each network layer in the first neural network model under the source framework and the reasoning result of the corresponding network layer in the second neural network model under the target framework may be compared sequentially so as to position a network layer which is precision-unaligned with the first neural network model earliest in the second neural network model, thus obtaining the target network layer.

[0121] At step S304, the model precision positioning apparatus outputs information of the target network layer.

[0122] After determining the target network layer, the model precision positioning apparatus may output the information of the target network layer. The information of the

target network layer includes an identifier of the target network layer, a parameter of the target network layer, output data of the target network layer, and an error between the output data and output data of a network layer corresponding to the target network layer in the first neural network model and the like. An administrator or user may analyze a cause of the precision un-alignment based on the information of the target network layer.

[0123] In the method of positioning a model precision according to an embodiment of the present disclosure, based on the output data of each network layer obtained by processing the same model input data through the first neural network model under the source framework and the second neural network model under the target framework, the precision-unaligned target network layer of the second neural network model may be positioned, thus precisely positioning the target network layer when the precision is not aligned.

[0124] FIG. 6 is another flowchart illustrating a method of positioning a model precision according to an embodiment of the present disclosure. The method includes the following steps.

[0125] At step S401, a model precision positioning apparatus obtains a first neural network model under a source framework, model input data, and information of a target framework.

[0126] The specific implementation of the step may be referred to step S101 of the embodiment of FIG. 1, or to step S201 of the embodiment of FIG. 3, or to step S301 of the embodiment of FIG. 5.

[0127] At step S402, the model precision positioning apparatus establishes a network topology by processing the model input data through the first neural network model, where the network topology includes information of network layers and a connection relationship of the network layers in the first neural network model.

[0128] The model precision positioning apparatus establishes the network topology by processing the model input data through the first neural network model, where the network topology includes information of network layers and a connection relationship of the network layers in the first neural network model. When each network layer of the first neural network model under the source framework is transformed to a format of the target framework respectively by using a neural network model transformation tool, the source framework may run a reasoning process one time on the first neural network model by using the model input data, such that information of the network layers and a connection relationship of the network layers in the first neural network model can be obtained. The above information of the network layers and connection relationship of the network layers

in the first neural network model form the network topology of the first neural network model.

[0129] At step S403, the model precision positioning apparatus obtains a second neural network model based on the network topology and the information of the target framework, records output data of each network layer of the first neural network model for processing the model input data, and establishes a mapping relationship between the network layers in the first neural network model and the second neural network model based on model transformation.

[0130] After the model precision positioning apparatus obtains the network topology of the first neural network model, the target framework has its own network structure definition and tool kit. The model precision positioning apparatus may obtain the second neural network model by performing adaptive network configuration for the network topology of the first neural network model using the network structure definition and/or the tool kit of the target framework based on the network topology of the first neural network model.

[0131] When each network layer of the first neural network model under the source framework is transformed to a format of the target framework respectively by using the neural network model transformation tool, the source framework may run a reasoning process one time on the first neural network model by using the model input data, such that information of the network layers and a connection relationship of the network layers in the first neural network model can be obtained. Next, the model precision positioning apparatus may obtain the second neural network model by performing adaptive configuration for the network topology of the first neural network model using the network structure definition and the tool kit of the target framework based on the network topology of the first neural network model. Furthermore, in a process of performing model transformation, a mapping relationship between the network layers in the first neural network model and the second neural network model may also be established. The mapping relationship between the network layers in the first neural network model and the second neural network model may be one to multiple, multiple to one or one to one, and correspondingly the target network layer may be one or more network layers.

[0132] At step S404, the model precision positioning apparatus obtains output data of each network layer of the second neural network model by processing the model input data through the second neural network model.

[0133] The specific implementation of this step may be referred to step S103 of the embodiment of FIG. 1, or to step S203 of the embodiment of FIG. 3.

[0134] In this embodiment, the output data of each network layer of the second neural network model is recorded, whereas in the embodiments of FIG. 1 or 3, the final reasoning result of the second neural network model, i.e. the output data of the last network layer of the second neural network model, is recorded.

[0135] At step S405, the model precision positioning apparatus determines a precision-unaligned target network layer in the second neural network model based on the output data of each network layer of the first neural network model and the second neural network model and the mapping relationship.

[0136] After obtaining the output data of each network layer in the first neural network model and the second neural network model and the mapping relationship between the network layers in the first neural network model and the second neural network model, the model precision positioning apparatus may determine a target network layer which is precision-unaligned earliest in the second neural network model based on the output data of each network layer in the first neural network model and the second neural network model and the mapping relationship.

[0137] Specifically, in a precision positioning process, to precisely position the target network layer which is precision-unaligned earliest, a sequence number is configured for each network layer of the first neural network model. The sequence numbers may be in an ascending order. Hence, in a case of positioning a precision, each network layer of the first neural network model may be sequentially called based on the sequence numbers of a plurality of network layers in the first neural network model.

[0138] For each called network layers, a reasoning result output by a previous network layer is input to a next network layer for reasoning so as to obtain a reasoning result of each network layer. Furthermore, the reasoning result of each network layer is locally and correspondingly stored according to the sequence number of each network layer. Further, the reasoning result of each network layer is also assigned an identifier. A plurality of inputs may be present on one network layer. Only the reasoning result output by each layer is stored because except for the initial input data, the input of other network layer is the output of a previous layer. Therefore, when the initial input and the output of each layer in the first neural network model are recorded, it means that the inputs and outputs of all layers are recorded. FIG. 4 is a schematic diagram illustrating a process of positioning a precision-

unaligned target network layer after transformation of the illustrated neural network model. The neural network model includes a plurality of network layers, such as convolutional layer, activation function, normalization layer, Eltwise layer and pooling layer and the like. After data is initially input to the first network layer of the neural network model, the output of each network layer is taken as an input of a next network layer. The reasoning result of the neural network model includes initial input data, and input/output data 1 to 10.

[0139] The second neural network model is obtained by transforming each network layer of the first neural network model to be under the target framework separately. The same model input data is input to the second neural network model and then a reasoning result of each layer of the second neural network model is obtained by using the input of each layer and stored locally. The sequence number of each network layer is taken as an index for storage. Of course, the reasoning process under the target framework also needs to be carried out in a new process.

[0140] After the reasoning result of each network layer of the neural network model under the target framework and the source framework is obtained, the reasoning result of each network layer in the first neural network model under the source framework and the reasoning result of the corresponding network layer in the second neural network model under the target framework may be compared sequentially based on the sequence number of each network layer so as to position a network layer which is precision-unaligned with the first neural network model earliest in the second neural network model, thus obtaining the target network layer.

[0141] At step S406, the model precision positioning apparatus outputs information of the target network layer.

[0142] The specific implementation of the step may be referred to step S304 of the embodiment of FIG. 5.

[0143] In the method of positioning a model precision according to an embodiment of the present disclosure, based on the output data of each network layer obtained by processing the same model input data through the first neural network model under the source framework and the second neural network model under the target framework, the precision-unaligned target network layer of the second neural network model may be positioned, thus precisely positioning the network layer when the precision is not aligned.

[0144] Based on the same idea as the above method of transforming a neural network model, as shown in FIG. 7, an embodiment of the present disclosure further provides an apparatus for transforming a model. The apparatus 1000 includes:

[0145] an obtaining unit 11, configured to obtain a first neural network model under a source framework, model input data, and information of a target framework;

[0146] a transforming unit 12, configured to obtain a second neural network model under the target framework by performing model transformation on the first neural network model under the source framework; and

[0147] a verifying unit 13, configured to obtain a precision verification result of the second neural network model based on a first reasoning result and a second reasoning result obtained by processing the model input data respectively through the first neural network model and the second neural network model.

[0148] In one implementation, the transforming unit 12 is configured to obtain the second neural network model under the target framework by performing model transformation on the first neural network model under the source framework through a first process;

[0149] the apparatus further includes:

[0150] a creating unit 14, configured to create a second process through the first process; and

[0151] a reasoning unit 15, configured to call the second process through the first process, such that the second process obtains the second reasoning result by executing the processing of the second neural network model for the model input data.

[0152] In another implementation, the verifying unit 13 is configured to, if a difference between the first reasoning result and the second reasoning result is within a preset error range, determine the precision verification result of the second neural network model as precision-aligned; and/or,

[0153] the verifying unit 13 is configured to, if the difference between the first reasoning result and the second reasoning result is outside the preset error range, determine the precision verification result of the second neural network model as precision-unaligned.

[0154] In another implementation, the apparatus further includes:

[0155] a positioning unit 16, configured to, in response to determining the precision verification result of the second neural network model as precision-unaligned, position a precision-unaligned target network layer in the second neural network model;

[0156] an outputting unit 17, configured to output information of the target network layer.

[0157] In another implementation, the positioning unit 16 includes:

[0158] an establishing unit 161, configured to establish a mapping relationship between network layers in the first neural network model and the second neural network model;

[0159] a determining unit 162, configured to determine the precision-unaligned target network layer in the second neural network model based on output data of each network layer obtained by processing the model input data respectively through the first neural network model and the second neural network model and the mapping relationship.

[0160] In another implementation, the model transformation apparatus re-performs model transformation on the first neural network model, wherein in the process of re-performing model transformation, output data of each network layer in the first neural network model is obtained, and/or, the mapping relationship between the network layers in the first neural network model and the second neural network model is established.

[0161] In another implementation, the transforming unit 12 is configured to: establish a network topology by processing the model input data through the first neural network model, where the network topology includes information of network layers and a connection relationship of the network layers in the first neural network model; and obtain the second neural network model based on the network topology and the information of the target framework.

[0162] The specific implementation of the above units may be referred to relevant descriptions of the method of transforming a neural network model shown in FIG. 1 or 3 and thus will not be repeated herein.

[0163] In the apparatus for transforming a model according to an embodiment of the present disclosure, based on the reasoning results obtained by processing the same model input data through the first neural network model under the source framework and the second neural network model under the target framework, the precision verification result of the second neural network model may be obtained automatically, thus improving the precision verification efficiency of the transformed neural network model.

[0164] Based on the same idea as the above method of positioning a model precision, as shown in FIG. 8, an embodiment of the present disclosure further provides an apparatus for positioning a model precision. The apparatus 2000 includes:

[0165] an obtaining unit 21, configured to obtain a first neural network model under a source framework, model input data and information of a target framework;

[0166] a transforming unit 22, configured to obtain a second neural network model under the target framework by performing model transformation on the first neural network model;

[0167] a positioning unit 23, configured to position a precision-unaligned target network layer in the second neural network model based on output data of each network layer obtained by processing the model input data respectively through the first neural network model and the second neural network model;

[0168] an outputting unit 24, configured to output information of the target network layer.

[0169] In one implementation, the information of the target network layer includes at least one of: an identifier of the target network layer, a parameter of the target network layer, output data of the target network layer, and an error between the output data and output data of a network layer corresponding to the target network layer in the first neural network model.

[0170] In another implementation, the apparatus further includes:

[0171] a first establishing unit 25, configured to establish a mapping relationship between network layers in the first neural network model and the second neural network model;

[0172] a positioning unit 23, configured to determine a precision-unaligned target network layer in the second neural network model based on the output data of each network layer in the first neural network model and the second neural network model and the mapping relationship.

[0173] In another implementation, the positioning unit 23 includes:

[0174] a calling unit 231, configured to compare the output data of the corresponding network layers in the first neural network model and the second neural network model sequentially based on the mapping relationship;

[0175] a determining unit 232, configured to, in response to determining that a difference between the output data of the corresponding network layers in the first neural network model and the second neural network model is outside a preset error range, determine the network layer in the corresponding network layers which belongs to the second neural network model as the target network layer.

[0176] In another implementation, the transforming unit 22 includes:

[0177] a second establishing unit 221, configured to establish a network topology by processing the model input data through the first neural network model, where the network topology includes information of network layers and a connection relationship of the network layers in the first neural network model; and

[0178] a second obtaining unit 222, configured to obtain the second neural network model based on the network topology and the information of the target framework.

[0179] The specific implementation of the above units may be referred to the specific descriptions of the embodiment of FIG. 5 or 6, and thus will not be repeated herein.

[0180] In the apparatus for positioning a model precision according to an embodiment of the present disclosure, based on the output data of each network layer obtained by processing the same model input data through the first neural network model under the source framework and the second neural network model under the target framework, the precision-unaligned target network layer of the second neural network model may be positioned, thus precisely positioning the network layer when the precision is not aligned.

[0181] As shown in FIG. 9, there is further provided a hardware structure of an electronic device. The electronic device is used to implement the methods of transforming a neural network model / positioning a model precision. Part or all of the above methods may be implemented by hardware or by software or firmware.

[0182] Optionally, the electronic device may be specifically implemented as chip or integrated circuit.

[0183] Optionally, when part or all of the methods of transforming a neural network model / positioning a model precision is implemented by software or firmware, this implementation may be carried out by the electronic device 3000 shown in FIG. 9. As shown in FIG. 9, the electronic device 3000 may include a memory 33 and a processor 34 (the processor 34 in the apparatus may be one or more, and one processor is illustrated in FIG. 9).

[0184] The electronic device 3000 may further include an input device 31 and an output device 32 as well as the memory 33 and the processor 34. In this embodiment, the input device 31, the output device 32, the memory 33 and the processor 34 may be connected with each other via a bus or in another manner. This connection is made by a bus as shown in FIG. 9.

[0185] The processor 34 is used to perform the steps of the methods shown in FIGS. 1, 3, 5, and 6.

[0186] Optionally, the programs for the implementation of the methods of transforming a neural network model / positioning a model precision may be stored in the memory 33. The memory 33 may be a physically separate unit or integrated with the processor 34. The memory 33 may also be used to store data.

[0187] Optionally, when part or all of the methods of transforming a neural network model / positioning a model precision is implemented by software, the electronic device may also only include the processor. The memory for storing the programs is located outside the electronic device. The processor is connected with the memory via a circuit or a wire to read and execute the programs in the memory.

[0188] The processor may be a central processing unit (CPU), a network processor (NP), or a wireless local area network (WLAN) device.

[0189] The processor may further include a hardware chip. The above hardware chip may be a application-specific integrated circuit (ASIC), a programmable logic device (PLD), or a combination thereof. The above PLD may be a complex programmable logic device (CPLD), a field-programmable gate array (FPGA) or a generic array logic (GAL), or any combination thereof.

[0190] The memory may include a volatile memory, for example, a random access memory (RAM); the memory may further include a non-volatile memory, for example, a flash memory, hard disk drive (HDD), or a solid-state drive (SSD); the memory may further include any combination of the above types of memories.

[0191] Persons skilled in the art shall understand that one or more embodiments of the present disclosure may be provided as methods, systems, or computer program products. Thus, one or more embodiments of the present disclosure may be adopted in the form of entire hardware embodiments, entire software embodiments or embodiments combining software and hardware. Further, one or more embodiments of the present disclosure may be adopted in the form of computer program products that are implemented on one or more computer available storage media (including but not limited to magnetic disk memory, CD-ROM, and optical memory and so on) including computer available program codes.

[0192] An embodiment of the present disclosure further provides a computer readable storage medium storing computer programs. The programs are executed by a processor to implement the steps of the method of transforming a neural network model according to any one embodiment of the present disclosure and/or the steps of the method of positioning a model precision according to any one embodiment of the present disclosure.

[0193] Different embodiments in the present disclosure are described in a progressive manner. Each embodiment focuses on the differences from other embodiments with those same or similar parts among the embodiments referred to each other. Particularly, since the apparatus embodiments are basically similar to the method embodiments, the apparatus

embodiments are briefly described with relevant parts referred to the descriptions of the method embodiments.

[0194] Specific embodiments of the present disclosure are described above. Other embodiments not described herein still fall within the scope of the appended claims. In some cases, the actions or steps recorded in the claims may be performed in a sequence different from the embodiments to achieve a desired result. Further, the processes shown in drawings do not necessarily require the shown particular sequence or continuous sequence to achieve the desired result. In some embodiments, multi-task processing and parallel processing are possible and may also be advantageous.

[0195] The embodiments of the subject and functional operations described in the present disclosure may be achieved in the following: a digital electronic circuit, a tangible computer software or firmware, a computer hardware including a structure disclosed in the present disclosure or a structural equivalent thereof, or a combination of one or more of the above. The embodiment of the subject described in the present disclosure may be implemented as one or more computer programs, that is, one or more modules in computer program instructions encoded on a tangible non-transitory program carrier for being executed by or controlling a data processing apparatus. Alternatively or additionally, the program instructions may be encoded on an artificially-generated transmission signal, such as a machine-generated electrical, optical or electromagnetic signal. The signal is generated to encode and transmit information to an appropriate receiver for execution by the data processing apparatus. The computer storage medium may be a machine-readable storage device, a machine readable storage substrate, a random or serial access memory device, or a combination of one or more of the above.

[0196] The processing and logic flows described in the present disclosure may be executed by one or more programmable computers executing one or more computer programs to perform operations based on input data and generate outputs to execute corresponding functions. The processing and logic flows may be further executed by a dedicated logic circuit, such as a field programmable gate array (FPGA) or an application specific integrated circuit (ASIC), and the apparatus may be further implemented as the dedicated logic circuit.

[0197] Computers suitable for executing computer programs may include, for example, a general-purpose and/or special-purpose microprocessor, or any other type of central processing unit. Generally, the central processing unit receives instructions and data from a read-only memory and/or random access memory. Basic components of a computer may

include a central processing unit for implementing or executing instructions and one or more storage devices for storing instructions and data. Generally, the computer may further include one or more mass storage devices for storing data, such as magnetic disks, magneto-optical disks or compact disks, or the computer is operably coupled to this mass storage device to receive data therefrom or transmit data thereto, or both. However, the computer does not necessarily have such device. In addition, the computer may be embedded in another device, such as a mobile phone, a personal digital assistant (PDA), a mobile audio or video player, a game console, a global positioning system (GPS) receiver, or a portable storage device, e.g., a universal serial bus (USB) flash drive, and so on.

[0198] Computer readable media suitable for storing computer program instructions and data may include all forms of non-volatile memories, media and memory devices, such as semi-conductor memory devices (e.g., Erasable Programmable Read-Only Memory (EPROM), Electrically Erasable Programmable Read-Only Memory (EEPROM) and flash memory device), magnetic disks (e.g., internal hard disk or removable disk), magneto-optical disks, and CD ROM and DVD-ROM disks. The processor and the memory may be supplemented by or incorporated into a dedicated logic circuit.

[0199] Although many specific implementation details are included in the present disclosure, these details should not be construed as limiting any scope of the present disclosure or the claimed scope, but are mainly used to describe the features of specific embodiments of the present disclosure. Certain features described in several embodiments of the present disclosure may also be implemented in combination in a single embodiment. On the other hand, various features described in the single embodiment may also be implemented separately or in any appropriate sub-combination in several embodiments. In addition, although the features may function in certain combinations as described above and even be initially claimed as such, one or more features from the claimed combination may be removed from the combination in some cases, and the claimed combination may refer to a sub-combination or a variation of the sub-combination.

[0200] Similarly, although the operations are described in a specific order in the drawings, this should not be understood as requiring these operations to be performed in the shown specific order or in sequence, or requiring all of the illustrated operations to be performed, so as to achieve a desired result. In some cases, multi-task processing and parallel processing may be advantageous. In addition, the separation of different system modules and components in the above embodiments should not be understood as requiring such

separation in all embodiments. Further, it is to be understood that the described program components and systems may be generally integrated together in a single software product or packaged into a plurality of software products.

[0201] Therefore, the specific embodiments of the subject are already described, and other embodiments are within the scope of the appended claims. In some cases, actions recorded in the claims may be performed in a different order to achieve the desired result. In addition, the processing described in the drawings is not necessarily performed in the shown specific order or in sequence, so as to achieve the desired result. In some implementations, multi-task processing and parallel processing may be advantageous.

[0202] The foregoing disclosure is merely illustrative of preferred embodiments of the present disclosure but not intended to limit the present disclosure, and any modifications, equivalent substitutions, adaptations thereof made within the spirit and principles of one or more embodiments in the present disclosure shall be encompassed in the scope of protection of one or more embodiments in the present disclosure.

CLAIMS

1. A method of transforming a neural network model, comprising:

obtaining, by a model transformation apparatus, a first neural network model under a source framework, model input data and information of a target framework;

obtaining, by the model transformation apparatus, a second neural network model under the target framework by performing model transformation on the first neural network model under the source framework; and

obtaining, by the model transformation apparatus, a precision verification result of the second neural network model based on a first reasoning result and a second reasoning result obtained by processing the model input data respectively through the first neural network model and the second neural network model.

2. The method of claim 1, wherein obtaining the second neural network model under the target framework by performing model transformation on the first neural network model under the source framework comprises:

obtaining the second neural network model under the target framework by performing, through a first process, model transformation on the first neural network model under the source framework;

wherein the method further comprises:

creating a second process through the first process; and

calling the second process through the first process in such a way that the second reasoning result is obtained by processing the model input data through the second neural network model in the second process.

3. The method of claim 1 or 2, wherein obtaining, by the model transformation apparatus, the precision verification result of the second neural network model based on the first reasoning result and the second reasoning result obtained by processing the model input data respectively through the first neural network model and the second neural network model comprises:

in response to determining that a difference between the first reasoning result and the second reasoning result is within a preset error range, determining the precision verification result of the second neural network model as precision-aligned, or

in response to determining that the difference between the first reasoning result and the second reasoning result is outside the preset error range, determining the precision verification result of the second neural network model as precision-unaligned.

4. The method of any one of claims 1 to 3, further comprising:

in response to determining the precision verification result of the second neural network model as precision-unaligned, positioning, by the model transformation apparatus, a precision-unaligned target network layer in the second neural network model; and

outputting, by the model transformation apparatus, information of the target network layer.

5. The method of claim 4, wherein positioning the precision-unaligned target network layer in the second neural network model comprises:

establishing, by the model transformation apparatus, a mapping relationship between network layers in the first neural network model and the second neural network model; and

determining, by the model transformation apparatus, the precision-unaligned target network layer in the second neural network model based on output data of each network layer obtained by processing the model input data respectively through the first neural network model and the second neural network model and the mapping relationship.

6. The method of claim 5, further comprising: re-performing, by the model transformation apparatus, model transformation on the first neural network model, wherein in the process of re-performing model transformation, output data of each network layer in the first neural network model is obtained, and/or, the mapping relationship between network layers in the first neural network model and the second neural network model is established.

7. The method of any one of claims 1 to 6, wherein obtaining, by the model transformation apparatus, the second neural network model under the target framework by performing model transformation on the first neural network model under the source framework comprises:

establishing, by the model transformation apparatus, a network topology by processing the model input data through the first neural network model, wherein the network topology

comprises information of the network layers and a connection relationship of the network layers in the first neural network model; and

obtaining, by the model transformation apparatus, the second neural network model based on the network topology and the information of the target framework.

8. A method of positioning a model precision, comprising:

obtaining, by a model precision positioning apparatus, a first neural network model under a source framework, model input data and information of a target framework;

obtaining, by the model precision positioning apparatus, a second neural network model under the target framework by performing model transformation on the first neural network model;

positioning, by the model precision positioning apparatus, a precision-unaligned target network layer in the second neural network model based on output data of each network layer obtained by processing the model input data respectively through the first neural network model and the second neural network model; and

outputting, by the model precision positioning apparatus, information of the target network layer.

9. The method of claim 8, further comprising:

establishing, by the model precision positioning apparatus, a mapping relationship between network layers in the first neural network model and the second neural network model based on the model transformation;

positioning, by the model precision positioning apparatus, the precision-unaligned target network layer in the second neural network model based on the output data of each network layer obtained by processing the model input data respectively through the first neural network model and the second neural network model comprises:

determining, by the model precision positioning apparatus, the precision-unaligned target network layer in the second neural network model based on the output data of each network layer in the first neural network model and the second neural network model and the mapping relationship.

10. The method of claim 9, wherein determining, by the model precision positioning apparatus, the precision-unaligned target network layer in the second neural network model

based on the output data of each network layer in the first neural network model and the second neural network model and the mapping relationship comprises:

comparing, by the model precision positioning apparatus, the output data of corresponding network layers in the first neural network model and the second neural network model sequentially based on the mapping relationship;

in response to determining that a difference between the output data of corresponding network layers in the first neural network model and the second neural network model is outside a preset error range, determining the network layer in the corresponding network layers which belongs to the second neural network model as the target network layer.

11. The method of any one of claims 8 to 10, wherein obtaining, by the model precision positioning apparatus, the second neural network model under the target framework by performing model transformation on the first neural network model under the source framework comprises:

establishing, by the model precision positioning apparatus, a network topology by processing the model input data through the first neural network model, wherein the network topology comprises information of the network layers and a connection relationship of the network layers in the first neural network model;

obtaining, by the model precision positioning apparatus, the second neural network model based on the network topology and the information of the target framework.

12. An apparatus for transforming a model, comprising:

an obtaining unit, configured to obtain a first neural network model under a source framework, model input data, and information of a target framework;

a transforming unit, configured to obtain a second neural network model under the target framework by performing model transformation on the first neural network model under the source framework; and

a verifying unit, configured to obtain a precision verification result of the second neural network model based on a first reasoning result and a second reasoning result obtained by processing the model input data respectively through the first neural network model and the second neural network model.

13. The apparatus of claim 12, wherein the transforming unit is configured to obtain the second neural network model under the target framework by performing model transformation on the first neural network model under the source framework through a first process;

the apparatus further comprises:

a creating unit, configured to create a second process through the first process; and

a reasoning unit, configured to call the second process through the first process, such that the second process obtains the second reasoning result by executing the processing of the second neural network model for the model input data.

14. The apparatus of claim 12 or 13, wherein the verifying unit is configured to, if a difference between the first reasoning result and the second reasoning result is within a preset error range, determine the precision verification result of the second neural network model as precision-aligned; and/or,

the verifying unit is configured to, in response to determining that the difference between the first reasoning result and the second reasoning result is outside the preset error range, determine the precision verification result of the second neural network model as precision-unaligned.

15. The apparatus of any one of claims 12-14, further comprising:

a positioning unit, configured to, in response to determining the precision verification result of the second neural network model as precision-unaligned, position a precision-unaligned target network layer in the second neural network model; and

an outputting unit, configured to output information of the target network layer.

16. The apparatus of claim 15, wherein the positioning unit comprises:

an establishing unit, configured to establish a mapping relationship between network layers in the first neural network model and the second neural network model; and

a determining unit, configured to determine the precision-unaligned target network layer in the second neural network model based on output data of each network layer obtained by processing the model input data respectively through the first neural network model and the second neural network model and the mapping relationship.

17. The apparatus of claim 16, wherein the model transformation apparatus re-performs model transformation on the first neural network model, wherein in the process of re-performing model transformation, output data of each network layer in the first neural network model is obtained, and/or, the mapping relationship between the network layers in the first neural network model and the second neural network model is established.

18. An apparatus for positioning a model precision, comprising:

- an obtaining unit, configured to obtain a first neural network model under a source framework, model input data and information of a target framework;

- a transforming unit, configured to obtain a second neural network model under the target framework by performing model transformation on the first neural network model;

- a positioning unit, configured to position a precision-unaligned target network layer in the second neural network model based on output data of each network layer obtained by processing the model input data respectively through the first neural network model and the second neural network model; and

- an outputting unit, configured to output information of the target network layer.

19. An electronic device, comprising a memory and a processor, wherein the memory stores program instructions, and the processor is configured to invoke the program instructions stored in the memory to implement the method according to any one of claims 1 to 11.

20. A computer storage medium, storing instructions, wherein the instructions cause, when run on a computer, the computer is caused to implement the method according to any one of claims 1 to 11.