

METHOD AND SYSTEM FOR TRAINING A GRAPH NEURAL NETWORK, AND METHOD OF IDENTIFYING AN ABNORMAL ACCOUNT

[0001] The present disclosure claims priority to Chinese Patent Application No. 202211316847.1, filed on October 26, 2022, and entitled ‘METHOD AND SYSTEM FOR TRAINING A GRAPH NEURAL NETWORK, AND METHOD OF IDENTIFYING AN ABNORMAL ACCOUNT’, which is incorporated herein by reference in its entirety.

FIELD

[0002] The present disclosure relates to the field of computer technologies, and in particular, to a method and system for training a graph neural network, and a method of identifying an abnormal account.

BACKGROUND

[0003] With the development of artificial intelligence technology, deep learning algorithms have achieved great success in application fields such as image processing and text processing. However, in reality, much of the data, such as social networks, knowledge graphs, and molecular structures, are often difficult to be processed with conventional deep learning algorithms due to its special data forms. In this regard, researchers have proposed various Graph Neural Networks (GNN) models for graph structure data, which have been widely used in content recommendation, drug research, chip design, and other fields.

[0004] In the related art, when training a graph neural network, graph structure data is generally processed first, and then training is performed with the processed graph structure data. The two steps are performed serially. However, since the time required for processing graph structure data and the time for training with processed graph structure data may not be consistent, it may result in that the graph structure data of the next round has not been processed. The graph structure data of the current round has been trained, or the graph structure data of the current round has been processed and the graph structure data of the previous round has not yet completed training. This in turn causes one of the processing steps to enter a state of waiting for execution. However, the processing step waiting for execution still occupies computing resources, affecting the utilization rate of computing resources.

SUMMARY

[0005] Embodiments of the present disclosure provide at least a method and system for training a graph neural network, and a method of identifying an abnormal account.

[0006] In a first aspect, an embodiment of the present disclosure provides a method of training a graph neural network, applied to any terminal device in a plurality of distributed training terminals for training a same target neural network, the method comprising:

[0007] obtaining initial graph structure data corresponding to the terminal device, the initial graph structure data respectively obtained by the plurality of distributed training terminals being derived from the same sample graph structure data; and

[0008] performing the following graph structure data processing stage and graph neural network training stage cyclically, until a target neural network satisfying a training requirement is obtained:

[0009] determining a processing opportunity for currently performing a graph structure data processing stage based on historical execution data of historically performing a graph structure data processing stage and a graph neural network training stage;

[0010] performing, based on the processing opportunity, graph structure data processing on the initial graph structure data in the graph structure data processing stage, to generate target graph structure data; the graph structure data processing comprising data sampling processing and feature extraction processing; and

[0011] training, based on the target graph structure data, the target neural network in the graph neural network training stage.

[0012] In a possible implementation, the initial graph structure data is allocated based on the following:

[0013] segmenting the sample graph structure data based on a breadth first search algorithm, to obtain a plurality of first graph structure data; and

[0014] allocating, from the plurality of first graph structure data, the initial graph structure data to the distributed training terminals based on a greedy algorithm, a number of terminals of the distributed training terminals, and a number of segments of the first graph structure data obtained by segmentation.

[0015] In a possible implementation, performing the graph structure data processing on the initial graph structure data comprises:

[0016] generating second graph structure data for characterizing a dependency relationship between respective operators based on an execution logic of a plurality of sampling processing and a plurality of feature extraction processing;

[0017] determining a topological order of the respective operators based on the second graph structure data; and

[0018] performing, based on the topological order of the respective operators, a plurality of sampling processing and a plurality of feature extraction processing on the initial graph structure data.

[0019] In a possible implementation, determining the processing opportunity for currently performing the graph structure data processing stage based on the historical execution data of historically performing the graph structure data processing stage and the graph neural network training stage comprises:

[0020] determining whether to perform the current graph structure data processing at a current time instant based on a performance parameter of the terminal device, a first historical average duration for historically performing the graph structure data processing stage, and a second historical average duration for historically performing the graph neural network training stage; and

[0021] in accordance with a determination of performing the current graph structure data processing at the current time instant, determining the processing opportunity as the current time instant; otherwise, redetermining whether to perform the current graph result data processing after a predetermined time interval.

[0022] In a possible implementation, training the target neural network based on the target graph structure data comprises:

[0023] inputting execution code corresponding to the graph neural network training stage into a target code compiler to obtain third graph structure data generated by the target code compiler after compiling the execution code; and

[0024] performing, based on the third graph structure data, data processing on the target graph structure data to implement training of the target neural network.

[0025] In a second aspect, an embodiment of the present disclosure provides a system of training a graph neural network, comprising a plurality of distributed training terminals and a database, wherein:

[0026] the distributed training terminals are configured to determine a processing opportunity for currently performing a graph structure data processing stage based on historical execution data of historically performing a graph structure data processing stage and a graph neural network training stage; perform, based on the processing opportunity, graph structure data processing on initial graph structure data in the graph structure data processing stage, to generate target graph structure data, the graph structure data processing comprises data sampling processing and feature extraction processing; train, based on the target graph structure data, a target neural network in the graph neural network training stage; and

[0027] the database is configured to store an initial graph data structure corresponding to each of the distributed training terminals and send the initial graph data structure to each of the distributed training terminals.

[0028] In a third aspect, an embodiment of the present disclosure further provides a method of identifying an abnormal account, comprising:

[0029] obtaining a historical browsing record corresponding to a to-be-verified account; and

[0030] determining, based on the historical browsing record and a trained target neural network, a target identification result for characterizing whether the to-be-verified account is an abnormal account; wherein the target neural network is trained based on any of the method of training the graph neural network of the first aspect.

[0031] In a possible implementation, determining, based on the historical browsing record and the trained target neural network, the target identification result for characterizing whether the to-be-verified account is the abnormal account comprises:

[0032] determining, based on the historical browsing record, browsing feature information corresponding to the to-be-verified account;

[0033] constructing, based on the browsing feature information, to-be-verified graph structure data corresponding to the to-be-verified account; and

[0034] inputting the to-be-verified graph structure data into the target neural network to obtain a target identification result output by the target neural network.

[0035] In a possible implementation, after determining the target identification result for characterizing whether the to-be-verified account is the abnormal account, the method further comprises:

[0036] in accordance with a determination that the to-be-verified account is the abnormal account, adjusting permission information corresponding to the to-be-verified account to perform permission control on the to-be-verified account.

[0037] In a fourth aspect, an embodiment of the present disclosure further provides an apparatus for training a graph neural network, applied to any terminal device in a plurality of distributed training terminals for training a same target neural network, and the apparatus comprises:

[0038] a first obtaining module configured to obtain initial graph structure data corresponding to the terminal device, the initial graph structure data respectively obtained by the plurality of distributed training terminals being derived from the same sample graph structure data; and

[0039] a cycling module configured to perform the following graph structure data processing stage and graph neural network training stage cyclically, until a target neural network satisfying a training requirement is obtained:

[0040] determining a processing opportunity for currently performing a graph structure data processing stage based on historical execution data of historically performing a graph structure data processing stage and a graph neural network training stage;

[0041] performing, based on the processing opportunity, graph structure data processing on the initial graph structure data in the graph structure data processing stage, to generate target graph structure data; the graph structure data processing comprising data sampling processing and feature extraction processing; and

[0042] training, based on the target graph structure data, the target neural network in the graph neural network training stage.

[0043] In a possible implementation, the initial graph structure data is allocated based on the following:

[0044] segmenting the sample graph structure data based on a breadth first search algorithm, to obtain a plurality of first graph structure data; and

[0045] allocating, from the plurality of first graph structure data, the initial graph structure data to the distributed training terminals based on a greedy algorithm, a number of terminals of the distributed training terminals, and a number of segments of the first graph structure data obtained by segmentation.

[0046] In a possible implementation, when performing the graph structure data processing on the initial graph structure data, the cycling module is configured to:

[0047] generate second graph structure data for characterizing a dependency relationship between respective operators based on an execution logic of a plurality of sampling processing and a plurality of feature extraction processing;

[0048] determine a topological order of the respective operators based on the second graph structure data; and

[0049] perform, based on the topological order of the respective operators, a plurality of sampling processing and a plurality of feature extraction processing on the initial graph structure data.

[0050] In a possible implementation, when determining the processing opportunity for currently performing the graph structure data processing stage based on the historical execution data of historically performing the graph structure data processing stage and the graph neural network training stage, the cycling module is configured to:

[0051] determine whether to perform the current graph structure data processing at a current time instant based on a performance parameter of the terminal device, a first historical average duration for historically performing the graph structure data processing stage, and a second historical average duration for historically performing the graph neural network training stage; and

[0052] in accordance with a determination of performing the current graph structure data processing at the current time instant, determine the processing opportunity as the current time instant; otherwise, redetermining whether to perform the current graph result data processing after a predetermined time interval.

[0053] In a possible implementation, when training the target neural network based on the target graph structure data, the cycling module is configured to:

[0054] input execution code corresponding to the graph neural network training stage into a target code compiler to obtain third graph structure data generated by the target code compiler after compiling the execution code; and

[0055] perform, based on the third graph structure data, data processing on the target graph structure data to implement training of the target neural network.

[0056] In a fifth aspect, an embodiment of the present disclosure further provides an apparatus for identifying an abnormal account, comprising:

[0057] a second obtaining module configured to obtain a historical browsing record corresponding to a to-be-verified account; and

[0058] a determining module configured to determine, based on the historical browsing record and a trained target neural network, a target identification result for characterizing whether the to-be-verified account is an abnormal account, wherein the target neural network is trained based on the method of training the graph neural network of the first aspect.

[0059] In a possible implementation, when determining, based on the historical browsing record and the trained target neural network, the target identification result for characterizing whether the to-be-verified account is the abnormal account, the determining module is configured to:

[0060] determine, based on the historical browsing record, browsing feature information corresponding to the to-be-verified account;

[0061] construct, based on the browsing feature information, to-be-verified graph structure data corresponding to the to-be-verified account; and

[0062] input the to-be-verified graph structure data into the target neural network to obtain a target identification result output by the target neural network.

[0063] In a possible implementation, after determining the target identification result for characterizing whether the to-be-verified account is the abnormal account, the determining module is further configured to:

[0064] in accordance with a determination that the to-be-verified account is the abnormal account, adjust permission information corresponding to the to-be-verified account to perform permission control on the to-be-verified account.

[0065] In a sixth aspect, an embodiment of the present disclosure further provides a computer device, , comprising: a processor, a memory, and a bus, the memory storing machine-readable instructions executable by the processor, when the computer device runs, the processor communicating with the memory through the bus, the machine-readable instructions, when executed by the processor, performing the steps of any of the possible implementations of the first aspect or the third aspect.

[0066] In a seventh aspect, an embodiment of the present disclosure further provides a computer-readable storage medium, wherein the computer-readable storage medium having a computer program stored thereon, the

computer program, when executed by a processor, performs the steps of any of the possible implementations of the first aspect or the third aspect.

[0067] The method and system for training a graph neural network, and a method of identifying an abnormal account provided in the embodiments of the present disclosure perform training of a graph neural network with a distributed training terminal. Herein, in the process of training the graph neural network, a processing opportunity for currently performing a graph structure data processing stage is determined based on historical execution data of historically performing a graph structure data processing stage and a graph neural network training stage, and graph structure data processing stage and graph neural network training stage are performed based on the determined processing opportunity. In this way, by controlling the processing opportunity of the graph structure data processing stage, allocation of computing resources to the graph structure data processing stage and the graph neural network training stage can be adjusted in real time, to implement reasonable allocation of computing resources. Further, by reasonably allocating computing resources to the graph structure data processing stage and the graph neural network training stage, a computing duration of the graph structure data processing stage can be the same as or similar to a computing duration of the graph neural network training stage, to prevent a certain training stage from entering a state of waiting, thereby improving the utilization rate of computing resources.

[0068] To make the above objectives, features, and advantages of the present disclosure more comprehensible, the following describes preferred embodiments in detail with reference to the accompanying drawings.

BRIEF DESCRIPTION OF THE DRAWINGS

[0069] In order to more clearly describe the technical solutions in the embodiments of the present disclosure, the drawings used in the descriptions of the embodiments will be briefly described below. The drawings in the following description are incorporated into and form a part of the specification. These drawings show the embodiments consistent with the present disclosure and are used together with the description to describe the technical solutions of the present disclosure. It should be understood that the following drawings show only some embodiments of the present disclosure, and therefore should not be considered as a limitation on the scope. For those of ordinary skill in the art, other related drawings can be obtained based on these drawings without creative efforts.

[0070] FIG. 1 is a flowchart of a method of training a graph neural network according to an embodiment of the present disclosure;

[0071] FIG. 2 is a schematic diagram of an architecture of a system of training a graph neural network in the method of training a graph neural network according to an embodiment of the present disclosure;

[0072] FIG. 3 is a flowchart of a method of identifying an abnormal account according to an embodiment of the present disclosure;

[0073] FIG. 4 is a schematic diagram of an architecture of an apparatus for training a graph neural network according to an embodiment of the present disclosure;

[0074] FIG. 5 is a schematic diagram of an architecture of an apparatus for identifying an abnormal account according to an embodiment of the present disclosure; and

[0075] FIG. 6 is a schematic diagram of a structure of a computer device according to an embodiment of the present disclosure.

DETAILED DESCRIPTION

[0076] To make the objectives, technical solutions, and advantages of the embodiments of the present disclosure clearer, the following clearly and completely describes the technical solutions in the embodiments of the present disclosure with reference to the accompanying drawings in the embodiments of the present disclosure. It is obvious that the described embodiments are some but not all of the embodiments of the present disclosure. Usually, components of the embodiments of the present disclosure described and shown in the accompanying drawings may be arranged and designed in various different configurations. Therefore, the detailed description of the embodiments of the present disclosure provided in the accompanying drawings is not intended to limit the scope of the claimed present disclosure, but merely characterizes selected embodiments of the present disclosure. Based on the embodiments of the present disclosure, all other embodiments obtained by a person of ordinary skill in the art without creative efforts fall within the protection scope of the present disclosure.

[0077] It should be noted that similar reference numerals and letters denote similar items in the following drawings. Therefore, once an item is defined in one drawing, it is not required to be further defined and explained in subsequent drawings.

[0078] The term ‘and/or’ herein is merely an association relationship describing that three relationships may exist. For example, A and/or B may represent the following three cases: A exists alone, both A and B exist, and B exists alone. In addition, the term ‘at least one of’ herein characterizes any of a plurality of or any combination of at least two of a plurality of. For example, at least one of A, B, and C may represent: any element selected from the set consisting of A, B, and C.

[0079] It may be understood that before using the technical solution disclosed in each embodiment of the present disclosure, a user shall be informed of the type, scope of use, and usage scenario of personal information involved in the present disclosure and the authorization of the user shall be obtained in an appropriate manner in accordance with relevant laws and regulations.

[0080] For example, in response to receiving an active request of the user, prompt information is sent to the user to explicitly prompt the user that the operation requested by the user needs to acquire and use the personal information of the user. Therefore, the user can independently choose whether to provide personal information to software or hardware such as an electronic device, an application, a server, or a storage medium that executes the operation of the technical solution of the present disclosure based on the prompt information.

[0081] As an optional but non-limiting implementation, in response to receiving an active request of the user, the prompt information may be sent to the user in a pop-up window, for example. The prompt information may be presented in the pop-up window in text. In addition, the pop-up window may further carry a selection control for the user to select ‘Agree’ or ‘Disagree’ to provide personal information to the electronic device.

[0082] It may be understood that the above notification and user authorization obtaining process is merely illustrative and does not constitute a limitation to the implementation of the present disclosure. Other methods that meet relevant laws and regulations may also be applied to the implementation of the present disclosure.

[0083] In the related art, in the process of training a graph neural network, a graph structure data processing stage and a graph neural network training stage are often performed serially, i.e., after the graph data processing is completed, the graph neural network training is performed based on the target graph structure data obtained after the completion of the graph data processing. However, this may result in idling of computing resources of the terminal device. In addition, even if the graph structure data processing stage and the graph neural network training stage are performed in parallel, the computing resources of the terminal device are limited. If the computing resources allocated to each processing stage are not balanced, it may result in that the graph structure data of the next round has not been processed. The graph structure data of the current round has been trained, or the graph structure data of the current round has been processed and the graph structure data of the previous round has not yet completed training. This in turn causes one of the processing steps to enter a state of waiting for execution, which may also result in idling of computing resources.

[0084] Based on the above research, the present disclosure provides a method and system for training a graph neural network, and a method of identifying an abnormal account perform training of a graph neural network with a distributed training terminal. Herein, in the process of training the graph neural network, a processing opportunity for currently performing a graph structure data processing stage is determined based on historical execution data of historically performing a graph structure data processing stage and a graph neural network training stage, and graph structure data processing stage and graph neural network training stage are performed based on the determined processing opportunity. In this way, by controlling the processing opportunity of the graph structure data processing stage, allocation of computing resources to the graph structure data processing stage and the graph neural network training stage can be adjusted in real time, to implement reasonable allocation of computing resources. Further, by reasonably allocating computing resources to the graph structure data processing stage and the graph neural network training stage, a computing duration of the graph structure data processing stage can be the same as or similar to a computing duration of the graph neural network training stage, to prevent a certain training stage from entering a state of waiting, thereby improving the utilization rate of computing resources.

[0085] For ease of understanding of this embodiment, a method of training a graph neural network disclosed in an embodiment of the present disclosure is first described in detail. An execution subject of the method of training a graph neural network provided in this embodiment of the present disclosure is generally a computer device with a certain computing capability. The computer device includes, for example, a terminal device, a server, or a further processing device. The terminal device may be a user equipment (UE), a mobile device, a user terminal, a terminal, a personal digital assistant (PDA), a handheld device, a computing device, an in-vehicle device, a wearable device, or the like. In some possible implementations, the method of training a graph neural network may be implemented by a processor invoking a computer-readable instruction stored in a memory.

[0086] FIG. 1 is a flowchart of a method of training a graph neural network according to an embodiment of the present disclosure. The method is applied to any terminal device in a plurality of distributed training terminals for training a same target neural network, the method comprising S101 and S102, wherein:

[0087] S101: Obtain initial graph structure data corresponding to the terminal device, the initial graph structure data respectively obtained by the plurality of distributed training terminals being derived from the same sample graph structure data.

[0088] S102: Perform the following graph structure data processing stage and graph neural network training stage cyclically, until a target neural network satisfying a training requirement is obtained.

[0089] Herein, the graph structure data processing stage includes the following:

[0090] S1021: Determine a processing opportunity for currently performing a graph structure data processing stage based on historical execution data of historically performing a graph structure data processing stage and a graph neural network training stage.

[0091] S1022: Perform, based on the processing opportunity, graph structure data processing on the initial graph structure data in the graph structure data processing stage, to generate target graph structure data; the graph structure data processing comprising data sampling processing and feature extraction processing.

[0092] The graph neural network training stage includes the following:

[0093] S1023: Train, based on the target graph structure data, the target neural network in the graph neural network training stage.

[0094] The following is a detailed description of the above steps.

[0095] For S101, the sample graph structure data is graph structure data for training the target neural network. The graph structure data consists of a Vertex and an Edge and may be configured to represent an association relationship between elements in fields such as protein analysis, chip design, and automatic driving. A network type of the target neural network may be a neural network capable of processing graph structure data, for example, a graph neural network (GNN) or a graph convolutional network (GCN).

[0096] In an actual application, in a distributed training scenario, initial graph structure data is allocated to each of the distributed training terminals, and each of the distributed training terminals performs distributed training based on the allocated graph structure data.

[0097] In a possible implementation, the initial graph structure data may be allocated through the following steps A1 and A2:

[0098] A1: Segment the sample graph structure data based on a breadth first search algorithm, to obtain a plurality of first graph structure data.

[0099] In this specification, the sample graph structure data may include a plurality of vertices and edges. The sample graph structure data including the plurality of vertices and edges may be divided into a plurality of first graph structure data based on a breadth first search algorithm.

[0100] A2: Allocate, from the plurality of first graph structure data, the initial graph structure data to the distributed training terminals based on a greedy algorithm, a number of terminals of the distributed training terminals, and a number of segments of the first graph structure data obtained by segmentation.

[0101] In this specification, the greedy algorithm may be configured to allocate the plurality of first graph structure data to each of the distributed training terminals, to reduce network data access overhead between each of the distributed training terminals in a subsequent data stage, thereby reducing network resource consumption caused by data transmission and improving the training efficiency of the target neural network.

[0102] Specifically, mathematical modeling may be performed based on the number of terminal devices of the distributed training terminal and a number of segments of the first graph structure data obtained by segmentation, to obtain a mathematical model for allocation by the greedy algorithm. Based on the greedy algorithm and the mathematical model, an allocation score is calculated for any first graph structure data, the

allocation score being configured to determine a distributed training terminal to which the first graph structure data is allocated. Herein, when calculating the allocation score of the first graph structure data, the allocation score may be calculated based on a type of a vertex included in the first graph structure data and a weight value corresponding to the type of the vertex. Different types of vertices may correspond to different weight values. The type of the vertex may include training, verification, testing, and the like.

[0103] For example, allocation for the first graph structure data ‘graph structure data 1’ is used as an example. Allocation scores of the ‘graph structure data 1’ for distributed training terminals 1 to 3 are score 1, score 2, and score 3, respectively, wherein $\text{score 1} < \text{score 2} < \text{score 3}$. Therefore, the ‘graph structure data 1’ may be allocated to the distributed training terminal 1 corresponding to the maximum allocation score based on the allocation scores.

[0104] In this way, through the greedy algorithm, a mathematical problem ‘How to allocate the plurality of first graph structure data obtained after the sample graph structure data is segmented to each of the distributed training terminals to achieve an optimal effect (i.e., reduce network resource consumption caused by data transmission)’ may be converted into ‘How to allocate any graph structure data to a proper distributed training terminal to achieve a local optimal effect (i.e., reduce network resource consumption caused by data transmission)’. By calculating an allocation score characterizing the matching degree of each of the first graph structure data and each of the distributed training terminals, the first graph structure data may be allocated to the distributed training terminal corresponding to the distributed training terminal having the highest allocation score to implement a local optimal effect. Therefore, a final allocation result may be obtained by combining the local optimal solutions.

[0105] For S102, the graph structure data processing may include sampling processing and feature extraction processing. For any distributed training terminal, during the sampling processing, sampling processing may be performed on the initial graph structure data obtained by the distributed training terminal, and sampling processing may also be performed on a further distributed training terminal other than the distributed training terminal based on an actual training requirement. During the feature extraction processing, feature extraction processing may be performed on graph structure data obtained after the sampling processing, to obtain target graph structure data that can be configured to train the target neural network.

[0106] In an actual application, in one graph structure data processing process, the sampling processing needs to be performed first, and then the feature extraction processing is performed. Therefore, the two steps are often performed serially in the related art. However, there may be no dependency relationship between different samplings (or feature extractions) in the graph structure data processing process. Therefore, the different samplings (or feature extractions) may also be performed concurrently, to improve the data processing efficiency of the graph structure data processing stage.

[0107] In a possible implementation, when the graph structure data processing is performed on the initial graph structure data, the following steps B1 to B3 may be performed:

[0108] B1: Generate second graph structure data for characterizing a dependency relationship between respective operators based on an execution logic of a plurality of sampling processing and a plurality of feature extraction processing.

[0109] B2: Determine a topological order of the respective operators based on the second graph structure data.

[0110] B3: Perform, based on the topological order of the respective operators, a plurality of sampling processing and a plurality of feature extraction processing on the initial graph structure data.

[0111] In this specification, the execution logic of the sampling processing and the feature extraction processing may be characterizing by execution code, and a programming language corresponding to the execution code may be, for example, a graph traversal language Gremlin.

[0112] Specifically, when generating the second graph structure data, the second graph structure data characterizing the dependency relationship between respective operators may be automatically constructed based on a sampling logic of the target neural network for a graph data structure, and a topological order characterizing the target processing training is determined based on the dependency relationship between respective operators in the second graph structure data.

[0113] In an actual application, since the terminal device can perform a plurality of graph structure data processes concurrently and the computing resources of the terminal device are limited, by adjusting a processing opportunity of the current graph structure data, a number of graph structure data processes executed at the same time instant may be changed, i.e., computing resources allocated to the graph structure data processing stage may be changed.

[0114] In a possible implementation, when determining the processing opportunity for currently performing the graph structure data processing stage based on the historical execution data of historically performing the graph structure data processing stage and the graph neural network training stage, whether to perform the current graph structure data processing at a current time instant may be determined based on a performance parameter of the terminal device, a first historical average duration for historically performing the graph structure data processing stage, and a second historical average duration for historically performing the graph neural network training stage.

[0115] In this specification, the performance parameter of the terminal device may include at least one of a CPU occupancy rate, an internal parameter occupancy rate, and a GPU occupancy rate. The first historical average duration of historically performing the graph structure data processing stage may be determined based on durations of each of the execution of the graph structure data processing stage before the current graph structure data processing stage is performed in the current training process and is configured to measure an amount of first computing resources allocated to the graph structure data processing stage in the current training process. The first historical average duration is inversely proportional to the first computing resources, i.e., the longer the first historical average duration is, the less the first computing resources are allocated to the graph structure data processing stage in the current training process. The second historical average duration for historically performing the graph neural network training stage may be determined based on durations of each of the execution of graph neural network training stage before the current graph neural network training stage is performed in the current training process and is configured to measure an amount of second computing resources allocated to the graph neural network training stage in the current training process. The second historical average duration is inversely proportional to the second computing resources, i.e., the longer the second historical average duration is, the less the second computing resources are allocated to the graph neural network training stage in the current training process.

[0116] Specifically, when determining whether to perform the current graph structure data processing at a current time instant based on a performance parameter of the terminal device, a first historical average duration for historically performing the graph structure data processing stage, and a second historical average duration for historically performing the graph neural network training stage, an execution score characterizing whether to perform current graph structure data processing at the current time instant may be determined based on a performance parameter of the terminal device, a first historical average duration for historically performing the graph structure data processing stage, and a second historical average duration for historically performing the graph neural network training stage. If the execution score is greater than a predetermined execution score threshold, it characterized that the current graph structure data processing needs to be performed at the current time instant. If the execution score is less than the predetermined execution score threshold, it characterized that the current graph structure data processing does not need to be performed at the current time instant.

[0117] Further, if the current graph structure data processing does not need to be performed at the current time instant, whether to perform current graph structure data processing at the current time instant (i.e., a time instant after the predetermined time interval) may be determined again based on a performance parameter of the terminal device, a first historical average duration for historically performing the graph structure data processing stage, and a second historical average duration for historically performing the graph neural network training stage again.

[0118] In this way, whether to perform current graph structure data processing at the current time instant may be determined based on the performance parameter of the terminal device at the current time instant the first historical average duration that can be configured to measure the first computing resources allocated to the graph structure data processing stage in the current training process, and the second historical average duration that can be configured to measure the second computing resources allocated to the graph neural network training stage in the current training process, so that the computing resources allocated to each processing stage can be controlled, thereby improving the training efficiency of the target neural network by reasonably allocating the computing resources.

[0119] Hereinafter, the content ‘determining whether to perform the current graph structure data processing at a current time instant’ in the embodiments of the present disclosure is described by way of example in combination with a specific calculation formula.

[0120] Specifically, depending on different tasks executed by the terminal device, a task processing module in the terminal device may be divided into a sampling module and a training module. In this specification, the sampling module is configured to execute a data processing task in the graph structure data processing stage. The training module is configured to execute a data processing task in the graph neural network training stage, the data processing task that needs to be executed may be stored in a form of a queue after being generated.

[0121] It is assumed that a performance parameter of the terminal device is A, a time interval from a previous execution of the graph structure data processing to the current time instant is T1, a length of a task queue of the sampling module is S, a first historical average duration for historically performing the graph structure data processing stage is T2, and a second historical average duration for historically performing the graph neural network training stage is T3. A calculation formula for determining an execution score X characterizing whether to perform current graph structure data processing at the current time instant may be:

$$\chi = f1 \times f2 \times T1$$

$$f1 = \frac{T2}{T3 \times S}$$

$$f2 = \alpha - e^{A/\beta}$$

[0122] Herein, both α and β are predetermined hyperparameters; f1 characterizes whether currently allocated computing resources of the terminal device for each processing stage are balanced; and f2 characterizes currently remaining computing resources of the terminal device.

[0123] In this way, by using the foregoing formula, the execution score characterizing whether to perform current graph structure data processing at the current time instant may be determined in combination with the currently allocated computing resources of the terminal device for each processing stage and the currently remaining computing resources. By comparing the execution score with a predetermined execution score threshold, a processing opportunity of performing the current graph structure data processing stage is determined. Thus, the currently allocated computing resources of the terminal device for each processing stage can be balanced, thereby improving the training efficiency of the target neural network.

[0124] In a possible implementation, when training the target neural network based on the target graph structure data, the following steps C1 and C2 may be performed.

[0125] C1: Input execution code corresponding to the graph neural network training stage into a target code compiler to obtain third graph structure data generated by the target code compiler after compiling the execution code.

[0126] Herein, execution code corresponding to the graph neural network training stage may be input into a target code compiler TensorFlow for compilation, to obtain the third graph structure data output by the target code compiler. The third graph structure data may be configured to represent an execution logic in the graph neural network training stage.

[0127] C2: Perform, based on the third graph structure data, data processing on the target graph structure data to implement training of the target neural network.

[0128] Herein, the target neural network may be trained based on an execution logic corresponding to the third graph structure data, the target graph structure data, and a predetermined graph neural network training algorithm. The embodiments of the present disclosure do not limit a specific type of the graph neural network training algorithm, provided that a target neural network that meets a predetermined training requirement can be obtained after implementation.

[0129] FIG. 2 is a schematic diagram of an architecture of a system of training a graph neural network according to an embodiment of the present disclosure. The system includes a plurality of distributed training terminals and a database, wherein:

[0130] The distributed training terminals are configured to determine a processing opportunity for currently performing a graph structure data processing stage based on historical execution data of historically performing a graph structure data processing stage and a graph neural network training stage; perform, based on the processing opportunity, graph structure data processing on initial graph structure data in the graph structure data processing stage, to generate target graph structure data, the graph structure data processing comprises data

sampling processing and feature extraction processing; train, based on the target graph structure data, a target neural network in the graph neural network training stage; and

[0131] the database is configured to store an initial graph data structure corresponding to each of the distributed training terminals and send the initial graph data structure to each of the distributed training terminals.

[0132] Specifically, the distributed training terminal may consist of a sampling module and a training module. The sampling module is configured to execute a data processing task in the graph structure data processing stage, and the training module is configured to execute a data processing task in the graph neural network training stage, the data processing task that needs to be executed may be stored in a form of a queue after being generated.

[0133] FIG. 3 is a flowchart of a method of identifying an abnormal account according to an embodiment of the present disclosure. The method includes S301 and S302, wherein:

[0134] S301: Obtain a historical browsing record corresponding to a to-be-verified account.

[0135] Herein, the historical browsing record corresponding to the to-be-verified account is a historical browsing record collected after a user authorizes.

[0136] S302: Determine, based on the historical browsing record and a trained target neural network, a target identification result for characterizing whether the to-be-verified account is an abnormal account; wherein the target neural network is trained based on the method of training the graph neural network of any embodiment of the present disclosure.

[0137] In a possible implementation, when determining, based on the historical browsing record and the trained target neural network, the target identification result for characterizing whether the to-be-verified account is the abnormal account, the following steps D1 to D3 may be used:

[0138] D1: Determine, based on the historical browsing record, browsing feature information corresponding to the to-be-verified account.

[0139] Herein, before the browsing feature information corresponding to the to-be-verified account is determined, data cleaning operation may also be performed on the historical browsing record.

[0140] D2: Construct, based on the browsing feature information, to-be-verified graph structure data corresponding to the to-be-verified account.

[0141] D3: Input the to-be-verified graph structure data into the target neural network to obtain a target identification result output by the target neural network.

[0142] In a possible implementation, after the target identification result indicating whether the to-be-verified account is an abnormal account is determined, permission information corresponding to the to-be-verified account may be adjusted when the to-be-verified account is an abnormal account, to perform permission control on the to-be-verified account.

[0143] Herein, adjusting the permission information corresponding to the to-be-verified account may be canceling a target permission corresponding to the to-be-verified account.

[0144] In addition, when the to-be-verified account is a normal account, the to-be-verified account may be verified again after a predetermined verification interval based on the predetermined verification interval.

[0145] Further, when the graph neural network trained by using the method of training a graph neural network according to the embodiment of the present disclosure is specifically applied, in addition to being applied to identifying an abnormal account, the graph neural network may also be applied to application scenarios such as

chip design, scene analysis and problem reasoning, content recommendation, drug research and development, traffic prediction for road traffic, and automatic driving. The embodiments of the present disclosure do not limit the specific application scenarios, provided that data processing in the application scenarios can be performed with the graph neural network.

[0146] For example, when the application scenario is content recommendation, browsing feature information of a target user may be determined based on a historical browsing record of the target user collected after the target user authorizes, so that content such as media content and an advertisement that are recommended for the target user may be determined based on the trained target neural network and the browsing feature information, so that the recommended content can meet different requirements of different users.

[0147] The method and system for training a graph neural network, and a method of identifying an abnormal account provided in the embodiments of the present disclosure perform training of a graph neural network with a distributed training terminal. Herein, in the process of training the graph neural network, a processing opportunity for currently performing a graph structure data processing stage is determined based on historical execution data of historically performing a graph structure data processing stage and a graph neural network training stage, and graph structure data processing stage and graph neural network training stage are performed based on the determined processing opportunity. In this way, by controlling the processing opportunity of the graph structure data processing stage, allocation of computing resources to the graph structure data processing stage and the graph neural network training stage can be adjusted in real time, to implement reasonable allocation of computing resources. Further, by reasonably allocating computing resources to the graph structure data processing stage and the graph neural network training stage, a computing duration of the graph structure data processing stage can be the same as or similar to a computing duration of the graph neural network training stage, to prevent a certain training stage from entering a state of waiting, thereby improving the utilization rate of computing resources.

[0148] A person skilled in the art may understand that, in the foregoing method of the specific implementation, an order of writing the steps does not mean a strict execution order and does not constitute any limitation on the implementation process. A specific execution order of the steps should be determined based on functions and a possible internal logic thereof.

[0149] Based on the same inventive concept, an embodiment of the present disclosure further provides an apparatus for training a graph neural network corresponding to the method of training a graph neural network. Because a principle of the apparatus in the embodiment of the present disclosure for solving a problem is similar to that of the method of training a graph neural network according to the embodiment of the present disclosure above, implementation of the apparatus may refer to implementation of the method, and details that are the same as those in the method will not be repeated.

[0150] FIG. 4 is a schematic diagram of an architecture of an apparatus for training a graph neural network according to embodiments of the present disclosure. The apparatus includes a first obtaining module 401 and a cycling module 402.

[0151] The first obtaining module 401 is configured to obtain initial graph structure data corresponding to the terminal device, the initial graph structure data respectively obtained by the plurality of distributed training terminals being derived from the same sample graph structure data; and

[0152] the cycling module 402 is configured to perform the following graph structure data processing stage and graph neural network training stage cyclically, until a target neural network satisfying a training requirement is obtained:

[0153] determining a processing opportunity for currently performing a graph structure data processing stage based on historical execution data of historically performing a graph structure data processing stage and a graph neural network training stage;

[0154] performing, based on the processing opportunity, graph structure data processing on the initial graph structure data in the graph structure data processing stage, to generate target graph structure data; the graph structure data processing comprising data sampling processing and feature extraction processing; and

[0155] training, based on the target graph structure data, the target neural network in the graph neural network training stage.

[0156] In a possible implementation, the initial graph structure data is allocated based on the following:

[0157] segmenting the sample graph structure data based on a breadth first search algorithm, to obtain a plurality of first graph structure data; and

[0158] allocating, from the plurality of first graph structure data, the initial graph structure data to the distributed training terminals based on a greedy algorithm, a number of terminals of the distributed training terminals, and a number of segments of the first graph structure data obtained by segmentation.

[0159] In a possible implementation, when performing the graph structure data processing on the initial graph structure data, the cycling module 402 is configured to:

[0160] generate second graph structure data for characterizing a dependency relationship between respective operators based on an execution logic of a plurality of sampling processing and a plurality of feature extraction processing;

[0161] determine a topological order of the respective operators based on the second graph structure data; and

[0162] perform, based on the topological order of the respective operators, a plurality of sampling processing and a plurality of feature extraction processing on the initial graph structure data.

[0163] In a possible implementation, when determining the processing opportunity for currently performing the graph structure data processing stage based on the historical execution data of historically performing the graph structure data processing stage and the graph neural network training stage, the cycling module 402 is configured to:

[0164] determine whether to perform the current graph structure data processing at a current time instant based on a performance parameter of the terminal device, a first historical average duration for historically performing the graph structure data processing stage, and a second historical average duration for historically performing the graph neural network training stage; and

[0165] in accordance with a determination of performing the current graph structure data processing at the current time instant, determine the processing opportunity as the current time instant; otherwise, redetermine whether to perform the current graph result data processing after a predetermined time interval.

[0166] In a possible implementation, when training the target neural network based on the target graph structure data, the cycling module 402 is configured to:

[0167] input execution code corresponding to the graph neural network training stage into a target code compiler to obtain third graph structure data generated by the target code compiler after compiling the execution code; and

[0168] perform, based on the third graph structure data, data processing on the target graph structure data to implement training of the target neural network.

[0169] The apparatus for training a graph neural network provided in the embodiments of the present disclosure performs training of a graph neural network with a distributed training terminal. Herein, in the process of training the graph neural network, a processing opportunity for currently performing a graph structure data processing stage is determined based on historical execution data of historically performing a graph structure data processing stage and a graph neural network training stage, and graph structure data processing stage and graph neural network training stage are performed based on the determined processing opportunity. In this way, by controlling the processing opportunity of the graph structure data processing stage, allocation of computing resources to the graph structure data processing stage and the graph neural network training stage can be adjusted in real time, to implement reasonable allocation of computing resources. Further, by reasonably allocating computing resources to the graph structure data processing stage and the graph neural network training stage, a computing duration of the graph structure data processing stage can be the same as or similar to a computing duration of the graph neural network training stage, to prevent a certain training stage from entering a state of waiting, thereby improving the utilization rate of computing resources.

[0170] FIG. 5 is a schematic diagram of an architecture of an apparatus for identifying an abnormal account according to embodiments of the present disclosure. The apparatus includes a second obtaining module 501 and a determining module 502.

[0171] The second obtaining module 501 is configured to obtain a historical browsing record corresponding to a to-be-verified account; and

[0172] the determining module 502 is configured to determine, based on the historical browsing record and a trained target neural network, a target identification result for characterizing whether the to-be-verified account is an abnormal account, wherein the target neural network is trained based on the method of training the graph neural network of any of the first aspect.

[0173] In a possible implementation, when determining, based on the historical browsing record and the trained target neural network, the target identification result for characterizing whether the to-be-verified account is the abnormal account, the determining module 502 is configured to:

[0174] determine, based on the historical browsing record, browsing feature information corresponding to the to-be-verified account;

[0175] construct, based on the browsing feature information, to-be-verified graph structure data corresponding to the to-be-verified account; and

[0176] input the to-be-verified graph structure data into the target neural network to obtain a target identification result output by the target neural network.

[0177] In a possible implementation, after determining the target identification result for characterizing whether the to-be-verified account is the abnormal account, the determining module 502 is further configured to:

[0178] in accordance with a determination that the to-be-verified account is the abnormal account, adjust permission information corresponding to the to-be-verified account to perform permission control on the to-be-verified account.

[0179] For a description of a processing process of each module in the apparatus and an interaction process between the modules, reference may be made to the related descriptions in the foregoing method embodiment, and details are not described herein again.

[0180] Based on the same technical concept, an embodiment of the present disclosure further provides a computer device. FIG. 6 is a schematic diagram of a structure of a computer device 600 according to embodiments of the present disclosure. The computer device 600 includes a processor 601, a memory 602, and a bus 603. The memory 602 is configured to store execution instructions and includes an internal memory 6021 and an external memory 6022. Herein, the internal memory 6021 is also referred to as an internal memory and is configured to temporarily store operation data in the processor 601 and data exchanged with an external memory 6022 such as a hard disk. The processor 601 exchanges data with the external memory 6022 through the internal memory 6021. When the computer device 600 runs, the processor 601 communicates with the memory 602 through the bus 603, so that the processor 601 executes the following instructions:

[0181] obtaining initial graph structure data corresponding to the terminal device, the initial graph structure data respectively obtained by the plurality of distributed training terminals being derived from the same sample graph structure data; and

[0182] performing the following graph structure data processing stage and graph neural network training stage cyclically, until a target neural network satisfying a training requirement is obtained:

[0183] determining a processing opportunity for currently performing a graph structure data processing stage based on historical execution data of historically performing a graph structure data processing stage and a graph neural network training stage;

[0184] performing, based on the processing opportunity, graph structure data processing on the initial graph structure data in the graph structure data processing stage, to generate target graph structure data; the graph structure data processing comprising data sampling processing and feature extraction processing; and

[0185] training, based on the target graph structure data, the target neural network in the graph neural network training stage.

[0186] In a possible implementation, in the instructions of the processor 601, the initial graph structure data is allocated based on the following:

[0187] segmenting the sample graph structure data based on a breadth first search algorithm, to obtain a plurality of first graph structure data; and

[0188] allocating, from the plurality of first graph structure data, the initial graph structure data to the distributed training terminals based on a greedy algorithm, a number of terminals of the distributed training terminals, and a number of segments of the first graph structure data obtained by segmentation.

[0189] In a possible implementation, in the instructions of the processor 601, performing the graph structure data processing on the initial graph structure data comprises:

[0190] generating second graph structure data for characterizing a dependency relationship between respective operators based on an execution logic of a plurality of sampling processing and a plurality of feature extraction processing;

[0191] determining a topological order of the respective operators based on the second graph structure data; and

[0192] performing, based on the topological order of the respective operators, a plurality of sampling processing and a plurality of feature extraction processing on the initial graph structure data.

[0193] In a possible implementation, in the instructions of the processor 601, determining the processing opportunity for currently performing the graph structure data processing stage based on the historical execution data of historically performing the graph structure data processing stage and the graph neural network training stage comprises:

[0194] determining whether to perform the current graph structure data processing at a current time instant based on a performance parameter of the terminal device, a first historical average duration for historically performing the graph structure data processing stage, and a second historical average duration for historically performing the graph neural network training stage; and

[0195] in accordance with a determination of performing the current graph structure data processing at the current time instant, determining the processing opportunity as the current time instant; otherwise, redetermining whether to perform the current graph result data processing after a predetermined time interval.

[0196] In a possible implementation, in the instructions of the processor 601, training the target neural network based on the target graph structure data comprises:

[0197] inputting execution code corresponding to the graph neural network training stage into a target code compiler to obtain third graph structure data generated by the target code compiler after compiling the execution code; and

[0198] performing, based on the third graph structure data, data processing on the target graph structure data to implement training of the target neural network; or

[0199] causing the processor 601 to execute the following instructions:

[0200] obtain a historical browsing record corresponding to a to-be-verified account; and

[0201] determine, based on the historical browsing record and a trained target neural network, a target identification result for characterizing whether the to-be-verified account is an abnormal account; wherein the target neural network is trained based on the method of training the graph neural network of any of the first aspect.

[0202] In a possible implementation, in the instructions of the processor 601, determining, based on the historical browsing record and the trained target neural network, the target identification result for characterizing whether the to-be-verified account is the abnormal account comprises:

[0203] determining, based on the historical browsing record, browsing feature information corresponding to the to-be-verified account;

[0204] constructing, based on the browsing feature information, to-be-verified graph structure data corresponding to the to-be-verified account; and

[0205] inputting the to-be-verified graph structure data into the target neural network to obtain a target identification result output by the target neural network.

[0206] In a possible implementation, in the instructions of the processor 601, after determining the target identification result for characterizing whether the to-be-verified account is the abnormal account, the method further comprises:

[0207] in accordance with a determination that the to-be-verified account is the abnormal account, adjusting permission information corresponding to the to-be-verified account to perform permission control on the to-be-verified account.

[0208] Embodiments of the present disclosure further provides a computer-readable storage medium, wherein the computer-readable storage medium having a computer program stored thereon, the computer program, when executed by a processor, performs the steps of the method of training a graph neural network described in the foregoing method embodiment. The storage medium may be a volatile or non-volatile computer-readable storage medium.

[0209] Embodiments of the present disclosure further provides a computer program product carrying program code, the program code comprising instructions that can be configured to perform the steps of the method of training a graph neural network described in the above method embodiment, which can be found in the above method embodiment, and which will not be further described herein.

[0210] Herein, the computer program product may be specifically implemented by hardware, software, or a combination thereof. In an optional embodiment, the computer program product is specifically embodied as a computer storage medium. In a further optional embodiment, the computer program product is specifically embodied as a software product, for example, a Software Development Kit (SDK), and the like.

[0211] Persons skilled in the art may clearly understand that, for the convenience and brevity of description, for a specific working process of the foregoing system and apparatus, reference may be made to corresponding processes in the foregoing method embodiments, and details are not described herein again. In the several embodiments provided in the present disclosure, it should be understood that the disclosed system, apparatus, and method may be implemented in other manners. The foregoing apparatus embodiments are merely illustrative. For example, the division of the units is merely a logical function division, and there may be a further division manner in actual implementation. For example, a plurality of units or components may be combined or integrated into a further system, or some features may be ignored or not performed. In addition, the displayed or discussed mutual couplings or direct couplings or communication connections may be implemented through some interfaces. The indirect couplings or communication connections between the apparatuses or units may be implemented in electrical, mechanical, or other forms.

[0212] The units described as separate parts may or may not be physically separate, and parts displayed as units may or may not be physical units, and may be located in one position, or may be distributed on a plurality of network units. Some or all of the units may be selected based on actual requirements to achieve the objectives of the solutions of the embodiments.

[0213] In addition, all the functional units in the embodiments of the present disclosure may be integrated into one processing unit, or each of the units may exist alone physically, or two or more units are integrated into one unit.

[0214] If the functions are implemented in the form of a software functional unit and sold or used as an independent product, the functions may be stored in a non-volatile computer-readable storage medium executable by a processor. Based on such an understanding, the technical solutions of the present disclosure are essentially, or the part contributing to the prior art, or the part of the technical solutions may be implemented in the form of a software product. The computer software product is stored in a storage medium and includes several instructions for instructing a computer device (which may be a personal computer, a server, a network device, or the like) to perform all or some of the steps of the method described in the embodiments of the present disclosure. The foregoing storage medium includes: any medium that can store program code, such as a USB flash drive, a mobile hard disk, a read-only memory (ROM), a random access memory (RAM), a magnetic disk, or an optical disc.

[0215] Finally, it should be noted that the foregoing embodiments are merely specific implementations of the present disclosure and are configured to describe the technical solutions of the present disclosure, but not to limit the present disclosure. The scope of protection of the present disclosure is not limited thereto. Although the present disclosure is described in detail with reference to the foregoing embodiments, persons of ordinary skill in the art should understand that: any person skilled in the art can still modify the technical solutions described in the foregoing embodiments or can easily conceive of changes to the technical solutions, or can make equivalent replacements to some technical features; however, these modifications, changes, and replacements do not cause the essence of the corresponding technical solutions to depart from the spirit and scope of the technical solutions of the embodiments of the present disclosure, and all should fall within the protection scope of the present disclosure. Therefore, the protection scope of the present disclosure shall be subject to the protection scope of the claims.

WHAT IS CLAIMED IS:

1. A method of training a graph neural network, applied to any terminal device in a plurality of distributed training terminals for training a same target neural network, the method comprising:

obtaining initial graph structure data corresponding to the terminal device, the initial graph structure data respectively obtained by the plurality of distributed training terminals being derived from the same sample graph structure data; and

performing the following graph structure data processing stage and graph neural network training stage cyclically, until a target neural network satisfying a training requirement is obtained:

determining a processing opportunity for currently performing a graph structure data processing stage based on historical execution data of historically performing a graph structure data processing stage and a graph neural network training stage;

performing, based on the processing opportunity, graph structure data processing on the initial graph structure data in the graph structure data processing stage, to generate target graph structure data; the graph structure data processing comprising data sampling processing and feature extraction processing; and

training, based on the target graph structure data, the target neural network in the graph neural network training stage.

2. The method of claim 1, wherein the initial graph structure data is allocated based on the following:

segmenting the sample graph structure data based on a breadth first search algorithm, to obtain a plurality of first graph structure data; and

allocating, from the plurality of first graph structure data, the initial graph structure data to the distributed training terminals based on a greedy algorithm, a number of terminals of the distributed training terminals, and a number of segments of the first graph structure data obtained by segmentation.

3. The method of claim 1, wherein performing the graph structure data processing on the initial graph structure data comprises:

generating second graph structure data for characterizing a dependency relationship between respective operators based on an execution logic of a plurality of sampling processing and a plurality of feature extraction processing;

determining a topological order of the respective operators based on the second graph structure data; and

performing, based on the topological order of the respective operators, a plurality of sampling processing and a plurality of feature extraction processing on the initial graph structure data.

4. The method of claim 3, wherein determining the processing opportunity for currently performing the graph structure data processing stage based on the historical execution data of historically performing the graph structure data processing stage and the graph neural network training stage comprises:

determining whether to perform the current graph structure data processing at a current time instant based on a performance parameter of the terminal device, a first historical average duration for historically performing

the graph structure data processing stage, and a second historical average duration for historically performing the graph neural network training stage; and

in accordance with a determination of performing the current graph structure data processing at the current time instant, determining the processing opportunity as the current time instant; otherwise, redetermining whether to perform the current graph result data processing after a predetermined time interval.

5. The method of claim 1, wherein training the target neural network based on the target graph structure data comprises:

inputting execution code corresponding to the graph neural network training stage into a target code compiler to obtain third graph structure data generated by the target code compiler after compiling the execution code; and

performing, based on the third graph structure data, data processing on the target graph structure data to implement training of the target neural network.

6. A system of training a graph neural network, comprising a plurality of distributed training terminals and a database, wherein:

the distributed training terminals are configured to determine a processing opportunity for currently performing a graph structure data processing stage based on historical execution data of historically performing a graph structure data processing stage and a graph neural network training stage; perform, based on the processing opportunity, graph structure data processing on initial graph structure data in the graph structure data processing stage, to generate target graph structure data, the graph structure data processing comprises data sampling processing and feature extraction processing; train, based on the target graph structure data, a target neural network in the graph neural network training stage; and

the database is configured to store an initial graph data structure corresponding to each of the distributed training terminals and send the initial graph data structure to each of the distributed training terminals.

7. A method of identifying an abnormal account, comprising:

obtaining a historical browsing record corresponding to a to-be-verified account; and

determining, based on the historical browsing record and a trained target neural network, a target identification result for characterizing whether the to-be-verified account is an abnormal account; wherein the target neural network is trained based on the method of training the graph neural network of any of claims 1 to 5.

8. The method of claim 7, wherein determining, based on the historical browsing record and the trained target neural network, the target identification result for characterizing whether the to-be-verified account is the abnormal account comprises:

determining, based on the historical browsing record, browsing feature information corresponding to the to-be-verified account;

constructing, based on the browsing feature information, to-be-verified graph structure data corresponding to the to-be-verified account; and

inputting the to-be-verified graph structure data into the target neural network to obtain a target identification result output by the target neural network.

9. The method of claim 7, wherein after determining the target identification result for characterizing whether the to-be-verified account is the abnormal account, the method further comprises:

in accordance with a determination that the to-be-verified account is the abnormal account, adjusting permission information corresponding to the to-be-verified account to perform permission control on the to-be-verified account.

10. An apparatus for training a graph neural network, applied to any terminal device in a plurality of distributed training terminals for training a same target neural network, and the apparatus comprises:

a first obtaining module configured to obtain initial graph structure data corresponding to the terminal device, the initial graph structure data respectively obtained by the plurality of distributed training terminals being derived from the same sample graph structure data; and

a cycling module configured to perform the following graph structure data processing stage and graph neural network training stage cyclically, until a target neural network satisfying a training requirement is obtained:

determining a processing opportunity for currently performing a graph structure data processing stage based on historical execution data of historically performing a graph structure data processing stage and a graph neural network training stage;

performing, based on the processing opportunity, graph structure data processing on the initial graph structure data in the graph structure data processing stage, to generate target graph structure data; the graph structure data processing comprising data sampling processing and feature extraction processing; and

training, based on the target graph structure data, the target neural network in the graph neural network training stage.

11. An apparatus for identifying an abnormal account, comprising:

a second obtaining module configured to obtain a historical browsing record corresponding to a to-be-verified account; and

a determining module configured to determine, based on the historical browsing record and a trained target neural network, a target identification result for characterizing whether the to-be-verified account is an abnormal account, wherein the target neural network is trained based on the method of training the graph neural network of any of claims 1 to 5.

12. A computer device, comprising: a processor, a memory, and a bus, the memory storing machine-readable instructions executable by the processor, when the computer device runs, the processor communicating with the memory through the bus, the machine-readable instructions, when executed by the processor, performing the

steps of the method of training a graph neural network of any of claims 1 to 5; or performing the steps of the method of identifying an abnormal account of any of claims 7 to 9.

13. A computer-readable storage medium, wherein the computer-readable storage medium having a computer program stored thereon, the computer program, when executed by a processor, performs the steps of the method of training a graph neural network of any of claims 1 to 5; or performs the steps of the method of identifying an abnormal account of any of claims 7 to 9.