DATA ANNOTATION METHOD AND RELATED DEVICE

[0001] This application claims priorities to Chinese Patent Application No. 202410361751.X, filed with the China National Intellectual Property Administration on March 27, 2024 and entitled "DATA ANNOTATION METHOD AND RELATED DEVICE", and to Chinese Patent Application No. 202410939017.7, filed with the China National Intellectual Property Administration on July 12, 2024 and entitled "DATA ANNOTATION METHOD AND RELATED DEVICE", both of which are incorporated herein by reference in their entireties.

TECHNICAL FIELD

[0002] This application relates to the field of artificial intelligence (artificial intelligence, AI) technologies, and in particular, to a data annotation method, an annotation system, a compute device cluster, a computer-readable storage medium, and a computer program product.

BACKGROUND

[0003] With the rapid development of AI technologies, AI agents (AI agent) have evolved into an independent technical approach. By encapsulating large models with specific functionalities, AI agents can support a broader range of application scenarios. For example, in game scenarios, users may define an AI agent for a non-player character (Non-player character), also referred to as an AI NPC. The AI NPC implements functionalities such as intelligent dialogues, chat-driven storylines, and settings of an NPC's background, personality, memory and emotional scoring by using the AI NPC.

[0004] AI agents such as AI NPCs may be implemented by using a scalable instructable multiworld agent (Scalable Instructable Multiworld Agent, SIMA). SIMA is capable of following natural language instructions to execute tasks across a plurality of games without requiring customized application programming interfaces (application programming interface, API). Furthermore, SIMA exhibits generalization capabilities, applying learned skills and strategies to unseen environments.

[0005] Inputs into SIMA may include screen images and simple natural language instructions provided by users. SIMA may map the images and the natural language instructions to keyboard operations and mouse operations (which may be collectively referred to as keyboard-and-mouse operations). However, training SIMA requires collecting, integrating, and annotation data of human experts such as video frames, natural language instructions, and keyboard-and-mouse

operation data. Manual collection and annotation of such data suffer from low efficiency and high cost, making it difficult to satisfy business demands.

SUMMARY

[0006] This application provides a data annotation method. In the method, a natural language instruction entered by a user, interactive operation data triggered through an interactive device, and corresponding interactive interface data during running of an application can be automatically collected, and different types of data are automatically aligned based on timestamps of all the data, thereby implementing automatic data annotation of an interactive task, and providing high-quality annotation data for training or testing an intelligent agent of the application. This application further provides a data annotation system, a compute device cluster, a computer-readable storage medium, and a computer program product that correspond to the method.

[0007] According to a first aspect, this application provides a data annotation method. The method may be performed by a data annotation system. The data annotation system is configured to annotate data. Annotation data may be used to learn of human operations and machine feedback, to train or test an agent such as an artificial intelligence agent (artificial intelligence agent, AI agent). The data annotation system may be a software system. The software system may be an independent software system, or a plugin or a functional module integrated into another software system. The software system may be deployed in a compute device cluster, and the compute device cluster executes program code of the software system, to perform the data annotation method in this application. The software system may be provided to a user in a form of a software package, and the user independently deploys the software package in a local compute device or a private cloud. Alternatively, the software system may be provided to a user in a form of a cloud service. For example, the software system may be deployed in a public cloud, and a cloud platform opens an application programming interface (application programming interface, API) of the cloud service for the user to use. In some possible implementations, the data annotation system may alternatively be a hardware system, for example, a compute device cluster having a data annotation capability. When the compute device cluster runs, the data annotation method in this application is performed.

[0008] Specifically, the data annotation system may obtain a natural language instruction entered by a user and interactive operation data triggered by the user through an interactive device, and obtain interactive interface data during running of an application. The interactive operation data includes an operation description of at least one interactive operation, and the interactive interface data includes an interactive interface video or a plurality of interactive interface images.

Then, the data annotation system extracts a task description of at least one task based on the natural language instruction. Then, the data annotation system associates a task description of a target task in the at least one task with a target interactive operation in the at least one interactive operation and a target interactive interface in the interactive interface data based on a timestamp of the natural language instruction, a timestamp of the interactive operation data, and a timestamp of the interactive interface data, to obtain annotation data. The target interactive interface is extracted from the interactive interface video or is at least one of the plurality of interactive interface images, and the annotation data is used to train or test an artificial intelligence AI agent.

[0009] In the method, the natural language instruction entered by the user, the interactive operation data triggered through the interactive device, and the corresponding interactive interface data during running of the application may be automatically collected, and different types of data are automatically aligned based on timestamps of all the data, thereby implementing automatic data annotation of an interactive task, and providing high-quality annotation data for training or testing an intelligent agent of the application. The annotation data may be directly used for training or testing, which reduces costs.

[0010] In some possible implementations, the data annotation system may further extract, based on the interactive interface video or the plurality of interactive interface images, interface change data caused by the at least one interactive operation. The data annotation system performs semantic recognition on the interface change data, to obtain an effect description of a task associated with the at least one interactive operation. Then, the data annotation system associates the task description of the target task in the at least one task with the target interactive operation in the at least one interactive operation, the target interactive interface in the interactive interface data, and an effect description of the target task.

[0011] In the method, the interface change data caused by the interactive operation is extracted, which can further improve accuracy of data annotation. In addition, the data annotation system further performs semantic recognition to obtain the effect description of the task, and associates the effect description of the task with the task description, the target interactive operation, and the target interactive interface, thereby increasing an amount of information of the annotation data, and providing richer information for training or testing the AI agent.

[0012] In some possible implementations, the user includes a plurality of users, and the plurality of users access the application through different channels. The data annotation system may associate the task description of the target task in the at least one task with the target interactive operation in the at least one interactive operation and the target interactive interface in the interactive interface data based on the timestamp and a channel identifier of the natural language instruction, the timestamp and a channel identifier of the interactive operation data, and

the timestamp and a channel identifier of the interactive interface data.

[0013] In this way, related data may be distinguished based on a channel identifier of the user, and fine-grained data annotation can be implemented. In this way, quality of the annotation data can be improved, and inference accuracy of an AI agent trained based on the annotation data is further improved.

[0014] In some possible implementations, the plurality of users access the application through different terminals, and the channel identifier includes a terminal identifier. In this way, data annotation in an application scenario in which a plurality of users interact with each other through respective terminals may be implemented, thereby meeting service requirements.

[0015] In some possible implementations, the plurality of users access the application through a same terminal. Interactive devices of the plurality of users are connected to the same terminal, and the channel identifier includes a device identifier of the interactive device. For example, in a multi-player online game scenario, a plurality of users may be connected to a same game console through respective gamepads to play a game. In this way, data annotation in an application scenario in which a plurality of users interact with each other through respective interactive devices may be implemented, thereby meeting service requirements.

[0016] In some possible implementations, the data annotation system may further obtain driving logic of the application. The data annotation system may associate the target interactive operation in the at least one interactive operation with the target interactive interface in the interactive interface data based on the timestamp of the interactive operation data and the timestamp of the interactive interface data, and then associate the task description of the target task in the at least one task with the target interactive operation or the target interactive interface based on the driving logic of the application.

[0017] In the method, data is associated by combining the driving logic of the application and the timestamp of the related data, which further improves accuracy of the annotation data.

[0018] In some possible implementations, the interactive operation includes at least one of a keyboard operation, a mouse operation, a screen touch operation, a motion sensing operation, or a controller operation. In this way, interactive data annotation for different interactive manners may be implemented. There is high availability and flexibility.

[0019] In some possible implementations, the application includes one or more of a game application, a design application, or a drawing application. The method may be applied to data annotation of general human-machine interactive tasks, instead of being limited to the game application, and has high availability.

[0020] According to a second aspect, this application provides a data annotation system. The data annotation system includes:

a collection subsystem, configured to: obtain a natural language instruction entered by a user and interactive operation data triggered by the user through an interactive device, and obtain interactive interface data during running of an application, where the interactive operation data includes an operation description of at least one interactive operation, and the interactive interface data includes an interactive interface video or a plurality of interactive interface images; and

an annotation subsystem, configured to extract a task description of at least one task based on the natural language instruction.

[0021] The annotation subsystem is further configured to associate a task description of a target task in the at least one task with a target interactive operation in the at least one interactive operation and a target interactive interface in the interactive interface data based on a timestamp of the natural language instruction, a timestamp of the interactive operation data, and a timestamp of the interactive interface data, to obtain annotation data, where the target interactive interface is extracted from the interactive interface video or is at least one of the plurality of interactive interface images, and the annotation data is used to train or test an artificial intelligence AI agent.

[0022] In some possible implementations, the annotation subsystem is further configured to:

extract, based on the interactive interface video or the plurality of interactive interface images, interface change data caused by the at least one interactive operation; and

perform semantic recognition on the interface change data, to obtain an effect description of a task associated with the at least one interactive operation; and

the annotation subsystem is specifically configured to:

associate the task description of the target task in the at least one task with the target interactive operation in the at least one interactive operation, the target interactive interface in the interactive interface data, and an effect description of the target task.

[0023] In some possible implementations, the user includes a plurality of users, and the plurality of users access the application through different channels; and

the annotation subsystem is specifically configured to:

associate the task description of the target task in the at least one task with the target interactive operation in the at least one interactive operation and the target interactive interface in the interactive interface data based on the timestamp and a channel identifier of the natural language instruction, the timestamp and a channel identifier of the interactive operation data, and the timestamp and a channel identifier of the interactive interface data.

[0024] In some possible implementations, the plurality of users access the application through different terminals, and the channel identifier includes a terminal identifier.

[0025] In some possible implementations, the plurality of users access the application through a same terminal, interactive devices of the plurality of users are connected to the same terminal,

and the channel identifier includes a device identifier of the interactive device.

[0026] In some possible implementations, the annotation subsystem is further configured to:

obtain driving logic of the application; and

the annotation subsystem is specifically configured to:

associate the target interactive operation in the at least one interactive operation with the target interactive interface in the interactive interface data based on the timestamp of the interactive operation data and the timestamp of the interactive interface data; and

associate the task description of the target task in the at least one task with the target interactive operation or the target interactive interface based on the driving logic of the application.

[0027] In some possible implementations, the interactive operation includes at least one of a keyboard operation, a mouse operation, a screen touch operation, a motion sensing operation, or a controller operation.

[0028] In some possible implementations, the application includes one or more of a game application, a design application, or a drawing application.

[0029] According to a third aspect, this application provides a compute device cluster. The compute device cluster includes at least one compute device, and the at least one compute device includes at least one processor and at least one memory. The at least one processor and the at least one memory communicate with each other. The at least one processor is configured to execute instructions stored in the at least one memory, to cause the compute device or the compute device cluster to perform the data annotation method according to any one of the first aspect or the implementations of the first aspect.

[0030] According to a fourth aspect, this application provides a computer-readable storage medium. The computer-readable storage medium stores instructions. The instructions instruct a compute device or a compute device cluster to perform the data annotation method according to any one of the first aspect or the implementations of the first aspect.

[0031] According to a fifth aspect, this application provides a computer program product including instructions. When the computer program product runs on a compute device or a compute device cluster, the compute device or the compute device cluster is caused to perform the data annotation method according to any one of the first aspect or the implementations of the first aspect.

[0032] In this application, the implementations provided in the foregoing aspects may be further combined to provide more implementations.

BRIEF DESCRIPTION OF DRAWINGS

[0033] To describe the technical methods in this application more clearly, the following briefly describes the required accompanying drawings.

[0034] FIG. 1 is a diagram of an architecture of a data annotation system according to this application;

[0035] FIG. 2 is a flowchart of a data annotation method according to this application;

[0036] FIG. 3 is a diagram of collected data according to this application;

[0037] FIG. 4 is a diagram of annotation data according to this application;

[0038] FIG. 5 is a schematic flowchart of a data annotation method in a multi-player online game scenario according to this application;

[0039] FIG. 6 is a schematic flowchart of generating a scenario in a design application according to this application;

[0040] FIG. 7 is a diagram of a structure of a compute device according to this application;

[0041] FIG. 8 is a diagram of a structure of a compute device cluster according to this application;

[0042] FIG. 9 is a diagram of a structure of another compute device cluster according to this application; and

[0043] FIG. 10 is a diagram of a structure of still another compute device cluster according to this application.

DESCRIPTION OF EMBODIMENTS

[0044] The terms "first" and "second" in embodiments of this application are merely intended for description, and shall not be understood as an indication or implication of relative importance or an implicit indication of a quantity of indicated technical features. Therefore, a feature limited by "first" or "second" may explicitly or implicitly include one or more features.

[0045] First, some technical terms in embodiments of this application are described.

[0046] An artificial intelligence agent (artificial intelligence agent, AI agent) may also be referred to as an intelligent agent, an intelligent subject, or an agent. In the AI field, the AI agent is an autonomous entity that senses an environment and takes an action (for example, performs an operation) to achieve a specific objective. Usually, the AI agent may be a software program, for example, independent software, or a functional module integrated into software. The AI agent may alternatively be a physical entity, for example, a compute device deployed with the foregoing software.

[0047] The AI agent may be widely applied to various fields, including but not limited to

robots, games, virtual assistants, autonomous vehicles, or next-generation application development platforms. The game field is used as an example. In a game scenario in which a non-player character (non-player character, NPC) is set, users may configure an AI NPC based on AI agent technologies, to implement functionalities such as intelligent dialogues, driven storylines, and settings of an NPC's background, personality, memory and emotions.

[0048] Typically, the AI agent may be trained based on massive annotation data. The game field is still used as an example for description. The AI agent may be a scalable instructable multiworld agent (Scalable Instructable Multiworld Agent, SIMA). Inputs into the SIMA may include screen images and simple natural language instructions provided by the users. The SIMA may map the images and the natural language instructions to keyboard and mouse operations. Based on this, the SIMA may be trained by annotating the natural language instructions and the screen images of the keyboard and mouse operations.

[0049] However, in related technologies, data of human experts such as video pictures, natural language instructions, and keyboard and mouse operation data are manually collected, integrated, and annotated. Manual collection and annotation of the data are inefficient and costly. Consequently, it is difficult to meet service requirements.

[0050] In view of this, this application provides a data annotation method. The method may be performed by a data annotation system. The data annotation system is configured to annotate data. Annotation data may be used to learn of human operations and machine feedback, to train an AI agent. A technology for learning of human operations and machine feedback may not only be applied to games, but also be a general agent development manner for a human-machine interactive task. Interactive operation data such as keyboard and mouse operations may be mapped to an interactive interface of a screen (for example, is interactive interface change data) to form annotation data, and the annotation data may be learned of to replace a task of human work.

[0051] The data annotation system may be a software system. The software system may be an independent software system, or a plugin or a functional module integrated into another software system. The software system may be deployed in a compute device cluster, and the compute device cluster executes program code of the software system, to perform the data annotation method in this application. The software system may be provided to a user in a form of a software package, and the user independently deploys the software package in a local compute device or a private cloud. Alternatively, the software system may be provided to the user in a form of a cloud service. For example, the software system may be deployed in a public cloud, and a cloud platform opens an application programming interface (application programming interface, API) of the cloud service for the user to use. In some possible implementations, the data annotation system may alternatively be a hardware system, for example, a compute device cluster having a data annotation

capability. When the compute device cluster runs, the data annotation method in this application is performed.

[0052] Specifically, the data annotation system may obtain a natural language instruction entered by a user and interactive operation data triggered by the user through an interactive device, and obtain interactive interface data during running of an application. The interactive operation data includes an operation description of at least one interactive operation, and the interactive interface data includes an interactive interface video or a plurality of interactive interface images. The data annotation system may extract a task description of at least one task based on the natural language instruction, and associate a task description of a target task in the at least one task with a target interactive operation in the at least one interactive operation and a target interactive interface in the interactive interface data based on a timestamp of the natural language instruction, a timestamp of the interactive operation data, and a timestamp of the interactive interface data, to obtain annotation data used to train or test an AI agent. The target interactive interface is extracted from the interactive interface video or is at least one of the plurality of interactive interface images.

[0053] In the method, the natural language instruction entered by the user, the interactive operation data triggered through the interactive device, and the corresponding interactive interface data during running of the application may be automatically collected, and different types of data are automatically aligned based on timestamps of all the data, thereby implementing automatic data annotation of an interactive task, and providing high-quality annotation data, for training or testing an intelligent agent of the application. The annotation data may be directly used for training or testing, which reduces costs.

[0054] To make the technical solutions in this application clearer and easier to understand, the following describes a system architecture of the data annotation system in this application with reference to the accompanying drawings.

[0055] FIG. 1 is a diagram of a system architecture of a data annotation system. A data annotation system 10 includes a collection subsystem 100 and an annotation subsystem 200. The collection subsystem 100 and the annotation subsystem 200 may be deployed in a cloud environment in a centralized or distributed manner. The cloud environment may be a cloud computing cluster formed by a plurality of compute devices. The cloud environment may include a public cloud, a private cloud, or a hybrid cloud. In some examples, the collection subsystem 100 and the annotation subsystem 200 may alternatively be deployed in a local compute device, for example, a local server. For ease of description, in FIG. 1, descriptions are provided by using an example in which the collection subsystem 100 and the annotation subsystem 200 are deployed in the cloud environment. The data annotation system 10 may provide, for a user to use, a cloud service (for example, a data annotation service) that provides a data annotation capability. For

example, the data annotation system 10 may open an API of the data annotation service for invocation.

[0056] Specifically, the collection subsystem 100 is configured to: obtain a natural language instruction entered by a user and interactive operation data triggered by the user through an interactive device, and obtain interactive interface data during running of an application. The natural language instruction may be an instruction based on a natural language, and the natural language instruction may include a text instruction or a voice instruction. The interactive operation data includes an operation description of at least one interactive operation, and the operation description may include an operation identifier (identifier, ID), for example, an operation name. The operation description may further include an operation parameter, for example, an operation duration. The operation description may further include a timestamp of triggering the interactive operation. The interactive interface data includes an interactive interface video or a plurality of interactive interface images. The interactive interface video may be a video obtained by performing screen recording on an interactive interface during running of an application, and the interactive interface image may be an image obtained by taking a screenshot of the interactive interface during running. As shown in FIG. 1, the collection subsystem 100 may obtain the interactive interface image by using a hook or through direct capturing in the application. For example, the collection subsystem 100 may directly write a plugin in a rendering module based on an unreal engine, or obtain a red green blue (red green blue, RGB) image, to obtain the interactive interface image. In this way, the interactive operation data, the natural language instruction (for example, a natural language instruction in a voice form, which may be briefly referred to as a voice instruction), and the interactive interface data (for example, an automatically captured image of a driven picture) may be automatically obtained, and dedicated manual annotation or recording is not required.

[0057] The annotation subsystem 200 is configured to: extract a task description of at least one task based on the natural language instruction, and then associate a task description of a target task in the at least one task with a target interactive operation in the at least one interactive operation and a target interactive interface in the interactive interface data based on a timestamp of the natural language instruction, a timestamp of the interactive operation data, and a timestamp of the interactive interface data, to obtain annotation data. The target interactive interface is extracted from the interactive interface video or is at least one of the plurality of interactive interface images. The annotation data is used to train or test an AI agent. A task, an operation, and a picture are automatically aligned by extracting a task description (for example, a task description, of a voice task, extracted from the voice instruction) and an interactive operation and picture frame list associated in a time sequence.

[0058] Further, the collection subsystem 100 and the annotation subsystem 200 may include a plurality of functional modules. The following describes in detail the functional modules of the collection subsystem 100 and the annotation subsystem 200.

[0059] In some possible implementations, the collection subsystem 100 may include an instruction collection module 102, an operation collection module 104, and an interface collection module 106. The instruction collection module 102 is configured to collect the natural language instruction. The operation collection module 104 is configured to collect the interactive operation data, for example, the operation description of the at least one interactive operation. The interface collection module 106 is configured to collect the interactive interface data, for example, the interactive interface video.

[0060] A communication software development kit (software development kit, SDK) and a driver layer may be deployed in a terminal. The driver layer may include a driver of an interactive device, for example, a driver of a keyboard or a driver of a mouse. The driver layer further includes a driver of a microphone. Code of the application may be deployed in the cloud environment. The terminal may access the application through the SDK and the driver layer. The user may trigger the interactive operation, for example, keyboard and mouse operations, through the communication SDK deployed in the terminal, or enter the natural language instruction in the voice form. The instruction collection module 102 may collect, based on the driver of the microphone in the driver layer, the natural language instruction in the voice form, and the operation collection module 104 may collect, based on the driver of the keyboard and the driver of the mouse in the driver layer, operation data of the keyboard and mouse operations. The operation data may be stored in an operation record (for example, an operation record file), and the natural language instruction in the voice form may be stored in an audio record (for example, an audio file). The interface collection module 106 may be connected to the application, to collect the interactive interface data. For example, the interface collection module 106 may obtain the interactive interface video by using a hook or through direct capturing in the application.

[0061] In some possible implementations, the annotation subsystem 200 may include a task extraction module 202 and an association module 204. The task extraction module 202 is configured to extract the task description of the at least one task based on the natural language instruction, and the association module 204 is configured to associate the task description of the target task in the at least one task with the target interactive operation in the at least one interactive operation and the target interactive interface in the interactive interface data based on the timestamp of the natural language instruction, the timestamp of the interactive operation data, and the timestamp of the interactive interface data, to obtain the annotation data.

[0062] Further, the annotation subsystem 200 may further include a mapping module 206. The

mapping module 206 is configured to map the operation to the interactive interface data. For example, the mapping module 206 may identify, from the interactive interface data, at least two frames of interactive interfaces with a picture change caused by the interactive operation, for example, an interactive interface before the change or a changed interactive interface, and map the interactive operation to the at least two frames of interactive interfaces. Further, the mapping module 206 may further generate an understanding of picture change content, and map the operation to the understanding of the picture change content.

[0063] It should be noted that, the data annotation system 10 in this application may implement automatic data collection and annotation for a computer operation application, to obtain annotation data. The annotation data may be input into an agent training system 20 to train an AI agent based on the annotation data. The agent training system 20 may be deployed in the cloud environment, to train the AI agent on the cloud. The AI agent may be built based on a transformer transformer. During training of the AI agent, a multi-modal vector embedding may be obtained, and the multi-modal embedding is aligned with the annotation data. The agent training system 20 may train the AI agent based on the multi-modal embedding. Further, the agent training system 20 may further play back a voice or execute a random task, to verify an interactive operation (action) and an effect correspondence.

[0064] It should be further noted that FIG. 1 shows merely a schematic division manner of the data annotation system 10. In another possible implementation of this embodiment of this application, the data annotation system 10 may be further divided into different subsystems or modules based on functions.

[0065] Based on the foregoing data annotation system 10, this application further provides a data annotation method. The following describes in detail a data annotation method in this application with reference to the accompanying drawings.

[0066] FIG. 2 is a flowchart of a data annotation method. The method includes the following steps.

[0067] S202: A data annotation system 10 obtains a natural language instruction entered by a user and interactive operation data triggered by the user through an interactive device.

[0068] The natural language instruction is an instruction issued in a natural language. A game application is used as an example. The natural language instruction may include "attack the tower". "Attack the tower" represents an act of attacking an enemy tower in a game. The natural language instruction may be an instruction in a voice form, or an instruction in a text form. For example, the user may send a voice of "attack the tower", or enter "attack the tower" in a text input box of an interactive interface.

[0069] The interactive operation data includes an operation description of at least one

interactive operation. The operation description may include an operation type. In some examples, the operation type may be classified based on the interactive device. Specifically, the interactive device may include a keyboard, a mouse, a motion sensing device, a controller, or a touchscreen. The controller may be a gamepad controller or a joystick controller. Correspondingly, the interactive operation may include at least one of a keyboard operation, a mouse operation, a screen touch operation, a motion sensing operation, or a controller operation. Based on this, the interactive operation data includes an operation type of an interactive operation triggered by the user through the interactive device. For example, the interactive operation data may be a keyboard A/virtual joystick/mouse.

[0070] In specific implementation, the data annotation system 10 initiates deployment of an environment of a cloud application (application, APP). The application may be a game application, a design application, or a drawing application. In some examples, when the data annotation system 10 is a cloud automatic annotation service, the data annotation system 10 may further initiate deployment of the associated collection subsystem 100 and the associated annotation subsystem 200. When a player accesses and operates, through a client (for example, a browser), an app deployed on the cloud, the data annotation system 10 may obtain the natural language instruction entered by the user and the interactive operation data triggered by the user through the interactive device. The data annotation system 10 may obtain, through a driver of a microphone, a natural language instruction in a voice form entered by the user, or obtain, through a driver of a keyboard or a mouse, a natural language instruction in a text form entered by the user.

[0071] It should be noted that the data annotation system 10 may present a license authorization interface to the user. The license authorization interface may include a license authorization control. The user (the player) may trigger the license authorization control after reading a license authorization agreement, to trigger a license authorization operation. Correspondingly, when obtaining user license authorization, the data annotation system 10 may obtain the natural language instruction entered by the user and the interactive operation data triggered by the user through the interactive device. An application developer may provide an executable program of the application to the data annotation system 10 (for example, a cloud automatic operation collection and data understanding service) on the cloud, to deploy the application in a cloud environment. The application developer may further select a specific user or user group (for example, a game player or a player group), to initiate automatic data collection and annotation (including data understanding) for the specific user or user group.

[0072] S204: The data annotation system 10 obtains interactive interface data during running of an application.

[0073] The interactive interface data includes an interactive interface video or a plurality of

interactive interface images. The data annotation system 10 may obtain the interactive interface video by performing screen recording on an interactive interface during running of the application, or obtain the interactive interface image by taking a screenshot of an interactive interface during running of the application. When the application is deployed in the cloud environment, the data annotation system 10 may obtain the interactive interface video or the plurality of interactive interface images during running of the application based on a cloud hosting environment. In some examples, the interactive interface image may be a video frame of a video. The video frame may be captured based on an operating system, or may be obtained by loading a plugin in underlying software such as an app or an engine.

[0074] In some possible implementations, the natural language instruction, the interactive operation data, and the interactive interface data may further include a timestamp (timestamp). The timestamp of the natural language instruction, the timestamp of the interactive operation data, and the timestamp of the interactive interface data may be used for data alignment and annotation.

[0075] The data annotation system 10 may store, based on the timestamp, the natural language instruction, the interactive operation data, and the interactive interface data that correspond to the timestamp. For ease of understanding, this application provides an example for description. As shown in FIG. 3, the data annotation system 10 may obtain a natural language instruction, interactive operation data, and interactive interface data that have a timestamp of 10:00:00. The natural language instruction includes audio with content "go to attack the tower, attack the tower", the interactive operation data may be "keyboard A/virtual joystick/mouse", and the interactive interface data may be video pictures during running of the application, and is specifically a frame n–1 and a frame n.

[0076] Further, for a multi-user collaborative application, for example, a multi-player online game, the data annotation system 10 may further obtain a channel identifier (channel ID) of data, each client connection may be considered as one channel, and data triggered or transmitted through the client connection may be associated with a user of a client. The data annotation system 10 may further implement user-grained data alignment based on the channel ID.

[0077] S206: The data annotation system 10 extracts a task description of at least one task based on the natural language instruction.

[0078] Specifically, when the natural language instruction is in a voice form, the data annotation system 10 may convert the natural language instruction in the voice form into a natural language instruction in a text form through speech to text (speech to text), and then the data annotation system 10 may extract the task description of the at least one task from the natural language instruction in the text form through text recognition. The task description may include a task type. For example, when the natural language instruction is "attack the tower", the data

annotation system 10 may extract the task description of the task based on the natural language instruction, which is specifically "attack the defense tower and crystal". In some possible implementations, the data annotation system 10 may alternatively directly extract the task description of the at least one task from the natural language instruction in the voice form. For example, the data annotation system 10 may train a task description extraction model based on a voice annotated with a task description, and the data annotation system 10 may input the natural language instruction in the voice form into the trained task description extraction model, to extract the task description of the at least one task.

[0079] When the natural language instruction is in the text form, the data annotation system 10 may extract the task description of the at least one task from the natural language instruction in the text form through text recognition. When the task description of the at least one task is extracted, extraction may be further performed based on prior knowledge. The prior knowledge may derive from service logic of the application or be manually customized. For example, in the game application, the prior knowledge may include that "tower" represents the defense tower.

[0080] S208: The data annotation system 10 associates a task description of a target task in the at least one task with a target interactive operation in the at least one interactive operation and a target interactive interface in the interactive interface data based on a timestamp of the natural language instruction, a timestamp of the interactive operation data, and a timestamp of the interactive interface data, to obtain annotation data.

[0081] The task description is extracted from the natural language instruction. The data annotation system 10 may associate the task description of the target task in the at least one task with the target interactive operation in the at least one interactive operation and the target interactive interface in the interactive interface data based on the timestamp of the natural language instruction, the timestamp of the interactive operation data, and the timestamp of the interactive interface data. The target interactive interface is extracted from the interactive interface video or is at least one of the plurality of interactive interface images. The task description of the target task may be extracted from a natural language instruction of a target time, and the target time matches the timestamp of the target interactive operation and the timestamp of the target interactive interface. In this application, that the timestamps are matched may be that the timestamps are equal or close, an absolute value of a difference between timestamps is less than a preset value, which indicates that the timestamp is close. The associated data is annotation data. The annotation data is used to train or test an AI agent.

[0082] In some possible implementations, the data annotation system 10 may further associate the target interactive operation in the at least one interactive operation with the target interactive interface in the interactive interface data based on the timestamp of the interactive operation data

and the timestamp of the interactive interface data, and associate the task description of the target task in the at least one task with the target interactive operation or the target interactive interface based on driving logic of the application. In this way, the task description of the target task in the at least one task may be associated with the target interactive operation in the at least one interactive operation and the target interactive interface in the interactive interface data, to obtain the annotation data. The driving logic defines a service flow, and the task and the interactive interface may be associated based on the driving logic. To ensure accuracy of annotation, the data annotation system 10 may associate the task description of the target task in the at least one task with the target interactive operation or the target interactive interface based on the timestamp and the driving logic of the application.

[0083] Further, the data annotation system 10 may further extract, based on the interactive interface video or the plurality of interactive interface images, interface change data caused by the at least one interactive operation. For example, the data annotation system 10 may obtain, based on a timestamp of the interactive operation, an interactive interface video corresponding to the timestamp, for example, an interactive interface video with a same timestamp or an interactive interface video of adjacent interfaces. Then, the data annotation system 10 may perform semantic analysis or semantic recognition on the interactive interface video, to extract an interactive interface image of a picture change caused by the interactive operation. The data annotation system 10 may process the interactive interface image, for example, perform difference processing, to obtain interface change data. The data annotation system 10 may perform semantic recognition on the interface change data, to obtain an effect description of a task associated with the at least one interactive operation. Correspondingly, the data annotation system 10 may associate the task description of the target task in the at least one task with the target interactive operation in the at least one interactive operation, the target interactive interface in the interactive interface data, and the effect description of the target task.

[0084] It should be noted that the data annotation system 10 may associate data with a specific timestamp, to implement automatic alignment of multi-dimensional data. In some examples, the data annotation system 10 may alternatively associate data of a period of time, to implement automatic alignment of multi-dimensional data.

[0085] As shown in FIG. 4, the following annotation data may be obtained by associating data with a timestamp of 10:00:00:

Data after data understanding: 10:00:00

Task task: attack the defense tower and crystal

Action associated operation: keyboard A/virtual joystick/mouse

Result associated result picture: frame n–1→frame n

Task effect understanding: one click on the keyboard A reduces the HP of the defense tower by 20%

[0086] The following annotation data may be obtained by associating data from a timestamp 10:00:00 to a timestamp 10:02:00:

Long-horizon task analysis (Long-Horizon): 10:00:00-10:02:00

Task task: go top and push the tower

Action associated operation: keyboard A/virtual joystick/mouse

Result associated result picture: frame n–1→frame n

Task effect understanding: one click on the keyboard A reduces the HP of the defense tower by 20%

[0087] Further, the data annotation system 10 may further input the annotation data into the AI agent for training, for example, fine-tune (fine-tuning) the AI agent. The application developer may query data content at any time, and initiate training or fine-tuning of the AI agent. When the application developer has an AI agent for the application, the data annotation system 10 may alternatively use the annotation data to verify validity of the AI agent.

[0088] In the method, the natural language instruction entered by the user, the interactive operation data triggered through the interactive device, and the corresponding interactive interface data during running of the application can be automatically collected, and different types of data are automatically aligned based on timestamps of all the data, thereby implementing automatic data annotation of the interactive task, and providing high-quality annotation data for training or testing an intelligent agent of the application. The annotation data may be directly used for training or testing, which reduces costs. In addition, the method can enhance adaptability of the application. For example, a domain-specific term such as "go top" in a game field may directly correspond to an operation and an interface, which has high availability.

[0089] Data such as the natural language instruction, the interactive operation data, and the interactive interface data during running of the application may be automatically collected under authorization of the application (for example, a game service) and an end user, thereby meeting security and privacy requirements.

[0090] The data annotation method shown in FIG. 2 may be applied to a single-person interactive application, or may be applied to a multi-player collaboration application. In the multi-player collaboration application, a plurality of users may access the application through different channels. The data annotation system 10 may alternatively associate the task description of the target task in the at least one task with the target interactive operation in the at least one interactive operation and the target interactive interface in the interactive interface data based on the timestamp and a channel identifier of the natural language instruction, the timestamp and a channel

identifier of the interactive operation data, and the timestamp and a channel identifier of the interactive interface data.

[0091] It should be noted that the plurality of users may access the application through different terminals. For example, in a multi-player online mobile game scenario, the plurality of users may access the application through respective smartphones. In this case, the channel identifier may be a terminal identifier. In some examples, the plurality of users may alternatively access the application through a same terminal. For example, in a multi-player online console game scenario, the plurality of users may access the application through a same terminal. The terminal may be, for example, a game console. Interactive devices of the plurality of users may be connected to the same terminal. The interactive devices of the plurality of users may be gamepad controllers of the plurality of users. In this case, the channel identifier may include a device identifier of the interactive device.

[0092] For ease of understanding, the following uses the multi-player online game scenario as an example to describe the data annotation method in this application.

[0093] FIG. 5 is a schematic flowchart of a data annotation method in a multi-player online game scenario. In this scenario, when a plurality of players are in a same game app, data of different players needs to be distinguished. An access connection (for example, a client connection) of each player is an independent channel, and keyboard and mouse operations and a sound of the player may be associated with a channel ID, to distinguish data of different users.

[0094] In this example, data of a user 1 is associated with a channel ID of a channel 1, and data of a user 2 is associated with a channel ID of a channel 2. The collection subsystem 100 in the data annotation system 10 deployed in the cloud environment may separately collect natural language instructions of different users, and store the natural language instructions in an audio record. The collection subsystem 100 further records an audio collection time. The collection subsystem 100 further separately collects interactive operation data of different users and timestamps, and stores the interactive operation data and the timestamps in an operation record. In addition, the collection subsystem 100 may further capture an interface during running of the application, and store the captured interface in a video frame record. The video frame record further records the timestamp of the interactive interface data. The annotation subsystem 200 in the data annotation system 10 may perform task extraction on the natural language instructions of different users to obtain task descriptions of tasks of different users, extract video information from video frame records of different users, and map video frames to operations. The annotation subsystem 200 may associate the tasks with the operations and the video frames, to obtain annotation data. For specific implementation of training or testing the AI agent based on the annotation data, refer to related content descriptions in the embodiment in FIG. 1. Details are not

described herein again.

[0095] An association relationship between the interactive interface data (a game screen) and an operation of a player may be implemented in a plurality of manners. One manner may be as follows: The data annotation system 10 obtains an association relationship between a channel and a player identifier (player ID) from a game application developer through a service interface or in a manual manner. When a picture changes, the data annotation system 10 may obtain a picture change situation of the player based on an image segmentation and recognition technology. Another manner may be as follows: A name of the player may be displayed in an interactive interface of a game application, and therefore, a mapping relationship between an operation of a player and an object operated by the player may be directly learned of by using a large vision model. The large vision model may learn, through text recognition and image recognition, the mapping relationship between an operation of a player and an object operated by the player.

[0096] The data annotation method in this application not only may be applied to the game application, but also may be generalized to all intelligent agent scenarios of human-machine interactive application software other than a game. For example, the data annotation method may be applied to a common desktop application. The following provides descriptions by using an example in which the desktop application is a design application. As shown in FIG. 6, when a home scene is generated, if lighting needs to be adjusted, for example, a user enters "change lighting of the scene to dim light", lighting may be usually controlled by integrating a Blender API and by using a Python script. Blender is open-source three-dimensional computer graphics software, and may be used to create and render a three-dimensional model, an animation, a game, a visual effect, and the like. Based on this, in the solution of this application, the data annotation system 10 may collect the natural language instruction, a mouse operation, and the interactive interface data of the user, perform automatic annotation to obtain annotation data, and train the AI agent based on the data. The AI agent may simulate the mouse operation, for example, simulate an operation of calling the Blender API by a user by using a mouse, to implement scene editing (including but not limited to generating the home scene).

[0097] Based on the foregoing content descriptions, it can be learned that the data annotation method in this application may implement automatic data collection and annotation for a computer operation application, reduce difficulty of data collection, improve accuracy, implement effective learning for a scene based on the annotation data, and may be generalized to all intelligent agent scenarios of human-machine interactive application software other than a game.

[0098] Based on the data annotation method in the embodiment shown in FIG. 2 to FIG. 6, this application further provides a data annotation system 10. As shown in FIG. 1, the data annotation system 10 includes:

a collection subsystem 100, configured to: obtain a natural language instruction entered by a user and interactive operation data triggered by the user through an interactive device, and obtain interactive interface data during running of an application, where the interactive operation data includes an operation description of at least one interactive operation, and the interactive interface data includes an interactive interface video or a plurality of interactive interface images; and

an annotation subsystem 200, configured to extract a task description of at least one task based on the natural language instruction.

[0099] The annotation subsystem 200 is further configured to associate a task description of a target task in the at least one task with a target interactive operation in the at least one interactive operation and a target interactive interface in the interactive interface data based on a timestamp of the natural language instruction, a timestamp of the interactive operation data, and a timestamp of the interactive interface data, to obtain annotation data, where the target interactive interface is extracted from the interactive interface video or is at least one of the plurality of interactive interface images, and the annotation data is used to train or test an artificial intelligence AI agent.

[00100] For example, the collection subsystem 100 and the annotation subsystem 200 may be implemented by hardware, or may be implemented by software.

[00101] When implemented by software, the collection subsystem 100 and the annotation subsystem 200 may be application programs running on a computer device, for example, a computing engine. The application program may also be virtualized, and provided to a user in a form of a virtualization service for use. The virtualization service may include a virtual machine (virtual machine, VM) service, a bare metal server (bare metal server, BMS) service, or a container (container) service. The VM service may be a service of virtualizing a virtual machine (virtual machine, VM) resource pool on a plurality of physical hosts by using a virtualization technology, to provide a VM on demand for the user to use. The BMS service is a service of virtualizing a BMS resource pool on a plurality of physical hosts to provide a BMS on demand for the user to use. The container service is a service of virtualizing a container resource pool on a plurality of physical hosts to provide a container on demand for the user to use. The VM is a simulated virtual computer, namely, a logical computer. The BMS is an elastically scalable high-performance computing service whose computing performance is the same as that of a conventional physical machine, and has a feature of secure physical isolation. The container is a kernel virtualization technology capable of providing lightweight virtualization to isolate user spaces, processes, and resources. It should be understood that the VM service, the BMS service, and the container service in the virtualization service are merely used as specific examples. During actual application, the virtualization service may alternatively be another lightweight or heavyweight virtualization

service. This is not specifically limited herein.

[00102] When implemented by hardware, the collection subsystem 100 and the annotation subsystem 200 may include at least one compute device, such as a server. Alternatively, the collection subsystem 100 and the annotation subsystem 200 may alternatively be devices implemented by using an application-specific integrated circuit (application-specific integrated circuit, ASIC) or a programmable logic device (programmable logic device, PLD). The PLD may be implemented by a complex programmable logic device (complex programmable logic device, CPLD), a field-programmable gate array (field-programmable gate array, FPGA), generic array logic (generic array logic, GAL), or any combination thereof.

[00103] In some possible implementations, the annotation subsystem 200 is further configured to:

extract, based on the interactive interface video or the plurality of interactive interface images, interface change data caused by the at least one interactive operation; and

perform semantic recognition on the interface change data, to obtain an effect description of a task associated with the at least one interactive operation.

[00104] The annotation subsystem 200 is specifically configured to:

associate the task description of the target task in the at least one task with the target interactive operation in the at least one interactive operation, the target interactive interface in the interactive interface data, and an effect description of the target task.

[00105] In some possible implementations, the user includes a plurality of users, and the plurality of users access the application through different channels.

[00106] The annotation subsystem 200 is specifically configured to:

associate the task description of the target task in the at least one task with the target interactive operation in the at least one interactive operation and the target interactive interface in the interactive interface data based on the timestamp and a channel identifier of the natural language instruction, the timestamp and a channel identifier of the interactive operation data, and the timestamp and a channel identifier of the interactive interface data.

[00107] In some possible implementations, the plurality of users access the application through different terminals, and the channel identifier includes a terminal identifier.

[00108] In some possible implementations, the plurality of users access the application through a same terminal, interactive devices of the plurality of users are connected to the same terminal, and the channel identifier includes a device identifier of the interactive device.

[00109] In some possible implementations, the annotation subsystem 200 is further configured to:

obtain driving logic of the application; and

the annotation subsystem 200 is specifically configured to:

associate the target interactive operation in the at least one interactive operation with the target interactive interface in the interactive interface data based on the timestamp of the interactive operation data and the timestamp of the interactive interface data; and

associate the task description of the target task in the at least one task with the target interactive operation or the target interactive interface based on the driving logic of the application.

[00110] In some possible implementations, the interactive operation includes at least one of a keyboard operation, a mouse operation, a screen touch operation, a motion sensing operation, or a controller operation.

[00111] In some possible implementations, the application includes one or more of a game application, a design application, or a drawing application.

[00112] This application further provides a compute device 700. As shown in FIG. 7, the compute device 700 includes a bus 702, a processor 704, a memory 706, and a communication interface 708. The processor 704, the memory 706, and the communication interface 708 communicate with each other through the bus 702. The compute device 700 may be a server or a terminal device. It should be understood that quantities of processors and memories in the compute device 700 are not limited in this application.

[00113] The bus 702 may be a peripheral component interconnect (peripheral component interconnect, PCI) bus, an extended industry standard architecture (extended industry standard architecture, EISA) bus, or the like. Buses may be classified into an address bus, a data bus, a control bus, and the like. For ease of representation, only one line is used to represent the bus in FIG. 7, but this does not mean that there is only one bus or only one type of bus. The bus 702 may include a path for transmitting information between components (for example, the memory 706, the processor 704, and the communication interface 708) of the compute device 700.

[00114] The processor 704 may include any one or more of processors such as a central processing unit (central processing unit, CPU), a graphics processing unit (graphics processing unit, GPU), a microprocessor (microprocessor, MP), or a digital signal processor (digital signal processor, DSP).

[00115] The memory 706 may include a volatile memory (volatile memory), for example, a random access memory (random access memory, RAM). The memory 706 may further include a non-volatile memory (non-volatile memory), for example, a read-only memory (read-only memory, ROM), a flash memory, a mechanical hard disk (hard disk drive, HDD), or a solid-state drive (solid-state drive, SSD). The memory 706 stores executable program code, and the processor 704 executes the executable program code to implement the foregoing data annotation method. Specifically, the memory 706 stores instructions used by the data annotation system 10 to perform

the data annotation method.

[00116] The communication interface 708 uses a transceiver module, for example, but not limited to, a network interface card or a transceiver, to implement communication between the compute device 700 and another device or a communication network.

[00117] An embodiment of this application further provides a compute device cluster. The compute device cluster includes at least one compute device. The compute device may be a server, for example, a central server, an edge server, or a local server in a local data center. In some embodiments, the compute device may alternatively be a terminal device, for example, a desktop computer, a notebook computer, or a smartphone.

[00118] As shown in FIG. 8, the compute device cluster includes at least one compute device 700. The memory 706 in one or more compute devices 700 in the compute device cluster may store same instructions used by the data annotation system 10 to perform the data annotation method.

[00119] In some possible implementations, the one or more compute devices 700 in the compute device cluster may alternatively be configured to execute a part of instructions used by the data annotation system 10 to perform the data annotation method. In other words, a combination of the one or more compute devices 700 may jointly execute the instructions used by the data annotation system 10 to perform the data annotation method.

[00120] It should be noted that the memory 706 in different compute devices 700 in the compute device cluster may store different instructions, to perform some functions of the data annotation system 10.

[00121] FIG. 9 shows a possible implementation. As shown in FIG. 9, two compute devices 700A and 700B are connected through a communication interface 708. A memory in the compute device 700A stores instructions used to perform functions of the collection subsystem 100 and the annotation subsystem 200. A memory in the compute device 700B stores instructions used to perform functions of the collection subsystem 100 and the annotation subsystem 200. In other words, the memories 706 in the compute devices 700A and 700B jointly store instructions used by the data annotation system 10 to perform the data annotation method.

[00122] A connection manner of the compute device cluster shown in FIG. 9 may consider that the data annotation method provided in this application requires a large quantity of computational power resources to analyze and associate collected data, to implement data annotation. Therefore, it is considered that a function implemented by the annotation subsystem 200 is performed by an independent compute device. For example, a function implemented by the collection subsystem 100 may be performed by the compute device 700A, and the function implemented by the annotation subsystem 200 is performed by the compute device 700B.

[00123] It should be understood that functions of the compute device 700A shown in FIG. 9 may alternatively be completed by a plurality of compute devices 700. Similarly, functions of the compute device 700B may alternatively be completed by a plurality of compute devices 700.

[00124] In some possible implementations, the one or more compute devices in the compute device cluster may be connected through a network. The network may be a wide area network, a local area network, or the like. FIG. 10 shows a possible implementation. As shown in FIG. 10, two compute devices 700C and 700D are connected through a network. Specifically, the compute devices are connected to the network through a communication interface in each compute device. In this type of possible implementation, the memory 706 in the compute device 700C stores instructions for executing a function of the collection subsystem 100. In addition, the memory 706 in the compute device 700D stores instructions for executing a function of the annotation subsystem 200.

[00125] A connection manner of a compute device cluster shown in FIG. 10 may consider that the data annotation method provided in this application requires a large quantity of compute resources for performing task description extraction and performing data association based on timestamps. Therefore, it is considered that a function implemented by the annotation subsystem 200 is performed by an independent compute device.

[00126] It should be understood that functions of the compute device 700C shown in FIG. 10 may alternatively be completed by a plurality of compute devices 700. Similarly, functions of the compute device 700D may alternatively be completed by a plurality of compute devices 700.

[00127] An embodiment of this application further provides a computer-readable storage medium. The computer-readable storage medium may be any usable medium that can be stored by a compute device, or a data storage device, such as a data center, including one or more usable media. The usable medium may be a magnetic medium (for example, a floppy disk, a hard disk, or a magnetic tape), an optical medium (for example, a DVD), a semiconductor medium (for example, a solid-state drive), or the like. The computer-readable storage medium includes instructions, and the instructions instruct the compute device to perform the data annotation method applied to the data annotation system 10.

[00128] An embodiment of this application further provides a computer program product including instructions. The computer program product may be software or a program product that includes instructions and that can run on a compute device or can be stored in any usable medium. When the computer program product runs on at least one compute device, the at least one compute device is caused to perform the foregoing data annotation method.

[00129] Finally, it should be noted that the foregoing embodiments are merely intended for describing the technical solutions of the present invention, but not for limiting the present

invention. Although the present invention is described in detail with reference to the foregoing embodiments, persons of ordinary skill in the art should understand that they may still make modifications to the technical solutions described in the foregoing embodiments or make equivalent replacements to some technical features thereof, without departing from the scope of the technical solutions of embodiments of the present invention.

CLAIMS

1. A data annotation method, wherein the method comprises:

obtaining a natural language instruction entered by a user and interactive operation data triggered by the user through an interactive device, and obtaining interactive interface data during running of an application, wherein the interactive operation data comprises an operation description of at least one interactive operation, and the interactive interface data comprises an interactive interface video or a plurality of interactive interface images;

extracting a task description of at least one task based on the natural language instruction; and

associating a task description of a target task in the at least one task with a target interactive operation in the at least one interactive operation and a target interactive interface in the interactive interface data based on a timestamp of the natural language instruction, a timestamp of the interactive operation data, and a timestamp of the interactive interface data, to obtain annotation data, wherein the target interactive interface is extracted from the interactive interface video or is at least one of the plurality of interactive interface images, and the annotation data is used to train or test an artificial intelligence AI agent.

2. The method according to claim 1, wherein the method further comprises:

extracting, based on the interactive interface video or the plurality of interactive interface images, interface change data caused by the at least one interactive operation; and

performing semantic recognition on the interface change data, to obtain an effect description of a task associated with the at least one interactive operation; and

the associating the task description of the target task in the at least one task with the target interactive operation in the at least one interactive operation and the target interactive interface in the interactive interface data comprises:

associating the task description of the target task in the at least one task with the target interactive operation in the at least one interactive operation, the target interactive interface in the interactive interface data, and an effect description of the target task.

3. The method according to claim 1 or 2, wherein the user comprises a plurality of users, and the plurality of users access the application through different channels; and

the associating the task description of the target task in the at least one task with the target interactive operation in the at least one interactive operation and the target interactive interface in the interactive interface data based on the timestamp of the natural language instruction, the timestamp of the interactive operation data, and the timestamp of the interactive interface data comprises:

associating the task description of the target task in the at least one task with the target interactive operation in the at least one interactive operation and the target interactive interface in the interactive interface data based on the timestamp and a channel identifier of the natural language instruction, the timestamp and a channel identifier of the interactive operation data, and the timestamp and a channel identifier of the interactive interface data.

4. The method according to claim 3, wherein the plurality of users access the application through different terminals, and the channel identifier comprises a terminal identifier.

5. The method according to claim 3, wherein the plurality of users access the application through a same terminal, interactive devices of the plurality of users are connected to the same terminal, and the channel identifier comprises a device identifier of the interactive device.

6. The method according to any one of claims 1 to 5, wherein the method further comprises:

obtaining driving logic of the application; and

the associating the task description of the target task in the at least one task with the target interactive operation in the at least one interactive operation and the target interactive interface in the interactive interface data comprises:

associating the target interactive operation in the at least one interactive operation with the target interactive interface in the interactive interface data based on the timestamp of the interactive operation data and the timestamp of the interactive interface data; and

associating the task description of the target task in the at least one task with the target interactive operation or the target interactive interface based on the driving logic of the application.

7. The method according to any one of claims 1 to 6, wherein the interactive operation comprises at least one of a keyboard operation, a mouse operation, a screen touch operation, a motion sensing operation, or a controller operation.

8. The method according to any one of claims 1 to 7, wherein the application comprises one or more of a game application, a design application, or a drawing application.

9. A data annotation system, wherein the system comprises:

a collection subsystem, configured to: obtain a natural language instruction entered by a user and interactive operation data triggered by the user through an interactive device, and obtain interactive interface data during running of the application, wherein the interactive operation data comprises an operation description of at least one interactive operation, and the interactive interface data comprises an interactive interface video or a plurality of interactive interface images; and

an annotation subsystem, configured to extract a task description of at least one task based on the natural language instruction, wherein

the annotation subsystem is further configured to associate a task description of a target task

in the at least one task with a target interactive operation in the at least one interactive operation and a target interactive interface in the interactive interface data based on a timestamp of the natural language instruction, a timestamp of the interactive operation data, and a timestamp of the interactive interface data, to obtain annotation data, wherein the target interactive interface is extracted from the interactive interface video or is at least one of the plurality of interactive interface images, and the annotation data is used to train or test an artificial intelligence AI agent.

10. The system according to claim 9, wherein the annotation subsystem is further configured to:

extract, based on the interactive interface video or the plurality of interactive interface images, interface change data caused by the at least one interactive operation; and

perform semantic recognition on the interface change data, to obtain an effect description of a task associated with the at least one interactive operation; and

the annotation subsystem is specifically configured to:

associate the task description of the target task in the at least one task with the target interactive operation in the at least one interactive operation, the target interactive interface in the interactive interface data, and an effect description of the target task.

11. The system according to claim 9 or 10, wherein the user comprises a plurality of users, and the plurality of users access the application through different channels; and

the annotation subsystem is specifically configured to:

associate the task description of the target task in the at least one task with the target interactive operation in the at least one interactive operation and the target interactive interface in the interactive interface data based on the timestamp and a channel identifier of the natural language instruction, the timestamp and a channel identifier of the interactive operation data, and the timestamp and a channel identifier of the interactive interface data.

12. The system according to claim 11, wherein the plurality of users access the application through different terminals, and the channel identifier comprises a terminal identifier.

13. The system according to claim 11, wherein the plurality of users access the application through a same terminal, interactive devices of the plurality of users are connected to the same terminal, and the channel identifier comprises a device identifier of the interactive device.

14. The system according to any one of claims 9 to 13, wherein the annotation subsystem is further configured to:

obtain driving logic of the application; and

the annotation subsystem is specifically configured to:

associate the target interactive operation in the at least one interactive operation with the target interactive interface in the interactive interface data based on the timestamp of the interactive

operation data and the timestamp of the interactive interface data; and

associate the task description of the target task in the at least one task with the target interactive operation or the target interactive interface based on the driving logic of the application.

15. The system according to any one of claims 9 to 14, wherein the interactive operation comprises at least one of a keyboard operation, a mouse operation, a screen touch operation, a motion sensing operation, or a controller operation.

16. The system according to any one of claims 9 to 15, wherein the application comprises one or more of a game application, a design application, or a drawing application.

17. A compute device cluster, wherein the compute device cluster comprises at least one compute device, the at least one compute device comprises at least one processor and at least one memory, the at least one memory stores computer-readable instructions, and the at least one processor executes the computer-readable instructions, to cause the compute device cluster to perform the data annotation method according to any one of claims 1 to 8.

18. A computer-readable storage medium, comprising computer-readable instructions, wherein the computer-readable instructions are used to implement the data annotation method according to any one of claims 1 to 8.

19. A computer program product, comprising computer-readable instructions, wherein the computer-readable instructions are used to implement the data annotation method according to any one of claims 1 to 8.