

ModelArts

SDK Reference

Issue 01
Date 2024-06-12



Copyright © Huawei Technologies Co., Ltd. 2024. All rights reserved.

No part of this document may be reproduced or transmitted in any form or by any means without prior written consent of Huawei Technologies Co., Ltd.

Trademarks and Permissions



HUAWEI and other Huawei trademarks are trademarks of Huawei Technologies Co., Ltd.

All other trademarks and trade names mentioned in this document are the property of their respective holders.

Notice

The purchased products, services and features are stipulated by the contract made between Huawei and the customer. All or part of the products, services and features described in this document may not be within the purchase scope or the usage scope. Unless otherwise specified in the contract, all statements, information, and recommendations in this document are provided "AS IS" without warranties, guarantees or representations of any kind, either express or implied.

The information in this document is subject to change without notice. Every effort has been made in the preparation of this document to ensure accuracy of the contents, but all statements, information, and recommendations in this document do not constitute a warranty of any kind, express or implied.

Huawei Technologies Co., Ltd.

Address: Huawei Industrial Base
Bantian, Longgang
Shenzhen 518129
People's Republic of China

Website: <https://www.huawei.com>

Email: support@huawei.com

Security Declaration

Vulnerability

Huawei's regulations on product vulnerability management are subject to the *Vul. Response Process*. For details about this process, visit the following web page:

<https://www.huawei.com/en/psirt/vul-response-process>

For vulnerability information, enterprise customers can visit the following web page:

<https://securitybulletin.huawei.com/enterprise/en/security-advisory>

Contents

1 Before You Start	1
2 SDK Overview	2
3 Getting Started	3
4 (Optional) Installing the ModelArts SDK Locally	4
5 Session Authentication	7
5.1 (Optional) Session Authentication.....	7
5.2 Authentication Using the Username and Password.....	8
5.3 AK/SK-based Authentication.....	8
6 OBS Management	10
6.1 Overview of OBS Management.....	10
6.2 Transferring Files (Recommended).....	10
6.3 Uploading a File to OBS.....	11
6.4 Uploading a Folder to OBS.....	12
6.5 Downloading a File from OBS.....	13
6.6 Downloading a Folder from OBS.....	14
7 Data Management	16
7.1 Managing Datasets.....	16
7.1.1 Querying a Dataset List.....	16
7.1.2 Creating a Dataset.....	17
7.1.3 Querying Details About a Dataset.....	28
7.1.4 Modifying a Dataset.....	28
7.1.5 Deleting a Dataset.....	29
7.2 Managing Dataset Versions.....	29
7.2.1 Obtaining a Dataset Version List.....	29
7.2.2 Creating a Dataset Version.....	30
7.2.3 Querying Details About a Dataset Version.....	31
7.2.4 Deleting a Dataset Version.....	32
7.3 Managing Samples.....	32
7.3.1 Querying a Sample List.....	32
7.3.2 Querying Details About a Sample.....	33
7.3.3 Deleting Samples in a Batch.....	33

7.4 Managing Dataset Import Tasks.....	34
7.4.1 Querying a Dataset Import Task List.....	34
7.4.2 Creating a Dataset Import Task.....	34
7.4.3 Querying the Status of a Dataset Import Task.....	37
7.5 Managing Export Tasks.....	38
7.5.1 Querying a Dataset Export Task List.....	38
7.5.2 Creating a Dataset Export Task.....	38
7.5.3 Querying the Status of a Dataset Export Task.....	39
7.6 Managing Manifest Files.....	39
7.6.1 Overview of Manifest Management.....	39
7.6.2 Parsing a Manifest File.....	40
7.6.3 Creating and Saving a Manifest File.....	42
7.6.4 Parsing a Pascal VOC File.....	43
7.6.5 Creating and Saving a Pascal VOC File.....	46
7.7 Managing Labeling Jobs.....	47
7.7.1 Creating a Labeling Job.....	47
7.7.2 Obtaining the Labeling Job List of a Dataset.....	48
7.7.3 Obtaining Details About a Labeling Job.....	49
8 Training Management.....	50
8.1 Training Jobs.....	50
8.1.1 Creating a Training Job.....	50
8.1.2 Debugging a Training Job.....	56
8.1.2.1 Using the SDK to Debug a Multi-Node Distributed Training Job.....	56
8.1.2.2 Using the SDK to Debug a Single-Node Training Job.....	59
8.1.3 Obtaining Training Jobs.....	63
8.1.4 Obtaining the Details About a Training Job.....	84
8.1.5 Modifying the Description of a Training Job.....	101
8.1.6 Deleting a Training Job.....	102
8.1.7 Terminating a Training Job.....	103
8.1.8 Obtaining Training Logs.....	120
8.1.9 Obtaining the Runtime Metrics of a Training Job.....	122
8.2 APIs for Resources and Engine Specifications.....	123
8.2.1 Obtaining Resource Flavors.....	123
8.2.2 Obtaining Engine Types.....	124
9 Model Management.....	126
9.1 Importing a Model.....	126
9.2 Obtaining Models.....	133
9.3 Obtaining Model Objects.....	135
9.4 Obtaining Details About a Model.....	137
9.5 Deleting a Model.....	140
10 Service Management.....	142

10.1 Service Management Overview.....	142
10.2 Deploying a Real-Time Service.....	142
10.3 Obtaining Details About a Service.....	152
10.4 Obtaining Services.....	155
10.5 Obtaining Service Objects.....	158
10.6 Updating Service Configurations.....	160
10.7 Obtaining Service Monitoring Information.....	164
10.8 Obtaining Service Logs.....	165
10.9 Delete a Service.....	167

1 Before You Start

This document describes how to install and configure a development environment and call functions provided by ModelArts SDK for secondary development.

Section	Description
SDK Overview	Concepts of ModelArts SDK
Getting Started	How to use ModelArts SDKs for secondary development
(Optional) Installing the ModelArts SDK Locally	How to install ModelArts SDKs locally
(Optional) Session Authentication	How to authenticate resources and initialize ModelArts SDK Client and OBS Client
Overview of OBS Management	How to call the SDK APIs of Object Storage Service (OBS), including the APIs for creating OBS buckets, uploading and downloading files and folders, as well as deleting OBS objects and buckets
ModelArts SDK operations: Data Management Model Management Service Management	Common operations using ModelArts SDK

2 SDK Overview

ModelArts Software Development Kits (ModelArts SDKs) encapsulate ModelArts REST APIs in Python language to simplify application development. You can directly call ModelArts SDKs to easily manage datasets, start AI training, generate models, and deploy the models as real-time services.

ModelArts SDKs support only Python of versions later than 3.7.x and earlier than 3.10.x. Version 3.7.x is recommended.

Scenarios

ModelArts SDKs can be used only in the ModelArts development environment notebook and local PC environment.

NOTICE

ModelArts SDKs cannot be used in training jobs or real-time services.

- ModelArts SDKs have been integrated into ModelArts notebook and can be directly used without session authentication.
Log in to the ModelArts management console, choose **DevEnviron > Notebook** in the navigation pane, create a notebook instance, and call the ModelArts SDKs on the terminal or IPYNB file. You can call SDKs on a notebook instance to perform operations such as OBS management, job management, model management, and service management by referring to the SDK reference.
- ModelArts SDKs can be installed, configured, and then used in local environments after session authentication.
 - a. Install the SDKs in a local path. Install the SDKs locally by referring to **(Optional) Installing the ModelArts SDK Locally**. If the SDKs have been installed in a local path, skip this step.
 - b. Perform session authentication by referring to **(Optional) Session Authentication**. The SDKs can be used after the authentication is complete.

3 Getting Started

ModelArts SDKs can be used only in the ModelArts development environment notebook and local PC environment.

NOTICE

ModelArts SDKs cannot be used in training jobs or real-time services.

- ModelArts SDKs have been integrated into ModelArts notebook and can be directly used without session authentication.
Log in to the ModelArts console, choose **DevEnviron** > **Notebook** in the navigation pane, create a notebook instance, and call the ModelArts SDKs on the terminal or IPYNB file. You can call SDKs on a notebook instance to perform operations such as OBS management, job management, model management, and service management by referring to the SDK reference.
- ModelArts SDKs can be installed, configured, and then used in local environments after session authentication.
 - a. Install the SDKs in a local path. Install the SDKs locally by referring to [\(Optional\) Installing the ModelArts SDK Locally](#). If the SDKs have been installed in a local path, skip this step.
 - b. Perform session authentication by referring to [\(Optional\) Session Authentication](#). The SDKs can be used after the authentication is complete.

4 (Optional) Installing the ModelArts SDK Locally

To use the ModelArts SDK on a PC or VM, install the ModelArts SDK in the target environment. After the installation, you can call the ModelArts SDK to easily manage datasets, create ModelArts training jobs and AI applications, and deploy the AI applications as real-time services.

Procedure

To install the ModelArts SDK locally, perform the following operations:

- [Step 1: Download the ModelArts SDK Package](#)
- [Step 2: Configure the Runtime Environment](#)
- [Step 3: Install the ModelArts SDK](#)

If an error occurred during the ModelArts SDK installation in Windows, rectify the fault by referring to [FAQ: An Error Occurred During ModelArts SDK Installation in Windows](#).

Step 1: Download the ModelArts SDK Package

Contact technical support to obtain the ModelArts SDK software package of the latest version.

Step 2: Configure the Runtime Environment

1. Check whether Python has been installed locally. If not, download Python of a proper version at the [Python official website](#) and install it. The Python version must be later than 3.7.x and earlier than 3.10.x. Version 3.7.x is recommended.

Run the **python --version** command in the local environment. If the following information is displayed, Python has been installed:

```
C:\Users\xxx>python --version  
Python 3.7.12
```

2. Check whether pip, package installer for Python, has been installed. If not, install pip by following the instructions provided at the [pip official website](#) after you install Python.

Run the **pip --version** command in the local environment. If the following information is displayed, pip has been installed:

```
C:\Users\xxx>pip --version
pip **.* from c:\users\xxx\appdata\local\programs\python\python**\lib\site-packages\pip (python *.*)
```

 **NOTE**

In Windows, if a message is displayed indicating that the command is not an internal or external command, add the Python and pip installation paths to **Path** in the environment variable. The pip installation path is typically the **Scripts** folder in the directory where Python is located.

1. Press **Win+R**, enter **sysdm.cpl** in the **Run** dialog box, and click **OK**.
 2. In the **System Properties** dialog box, click the **Advanced** tab and click **Environment Variables**.
 3. In the **User variables for** area, double-click **Path**. In the **Edit environment variable** dialog box, click **New** and add the Python and pip installation paths. The installation path must point to the **Scripts** folder, for example, **C:\python\python**\Scripts**.
3. Configure the pip source. The following uses Windows as an example to describe how to configure the pip source:
- a. Create a **pip** folder. Start **cmd** and run the **set** command to view the **AppData** path. Create a **pip** folder in the obtained **AppData** path. An example is provided as follows:


```
C:\Users\xxx>set
ALLUSERSPROFILE=C:\ProgramData
APPDATA=C:\Users\xxx\AppData\Roaming
```

The preceding information indicates that the **pip** folder needs to be created in **C:\Users\xxx\AppData\Roaming**.
 - b. Create a text file named **pip** in the **pip** folder and change the file name extension from **.txt** to **.ini**. An example is provided as follows:

index-url is the IP address of the pip source, which needs to be replaced as required.

```
[global]
index-url = https://pip.xxxx.com/repository/pypi/simple
trusted-host = pip.xxxx.com
disable-pip-version-check = true
timeout = 120
[install]
ignore-installed = true
no-dependencies = yes
```
4. Start **cmd** and run the following command to download the package of the required pip source:
- ```
C:\Users\xxx>pip install numpy # Replace numpy with the package you want to download.
```

### Step 3: Install the ModelArts SDK

Start **cmd** and run the following command to install the ModelArts SDK:

**pip install {Path to the SDK software package}\modelarts-latest-py2.py3-none-any.whl**

```
C:\Users\xxx>pip install C:\Users\xxx\Downloads\modelarts-latest-py2.py3-none-any.whl
.....
Successfully installed Pillow-*.0 ... modelarts-*** ...
```

When SDK is installed, dependency packages are installed by default. If message "Successfully installed" is displayed, the ModelArts SDK has been installed.



# 5 Session Authentication

## 5.1 (Optional) Session Authentication

### Overview

The session module authenticates resources and initializes ModelArts SDK Client and OBS Client. After a session is set up, you can directly call the ModelArts SDKs.

- ModelArts notebook does not require session authentication. The sample code is as follows:

```
from modelarts.session import Session
session = Session()
```
- Session authentication is required when the local PC uses ModelArts SDKs. You can select either of the following authentication modes:
  - **Authentication Using the Username and Password:** Available for **OBS Management, Data Management, Training Management, Model Management,** and **Service Management.**
  - **AK/SK-based Authentication:** Available for **OBS Management, Data Management, Training Management, Model Management,** and **Service Management.**

### Authentication Using the Username and Password

After installing the ModelArts SDK on the local PC, you can perform session authentication using the username and password. The sample code is as follows:

- Authentication using an account

Set **username** to your account name.

```
from modelarts.session import Session

Hardcoded or plaintext password is risky. For security, encrypt your password and store it in the
configuration file or environment variables.
In this example, the password is stored in environment variables for identity authentication. Before
running this example, set environment variable HUAWEICLOUD_SDK_PASSWORD.
__PASSWORD = os.environ["HUAWEICLOUD_SDK_PASSWORD"]
Decrypt the password if it is encrypted.
session = Session(username='****', password=__PASSWORD, region_name='****', project_id='****')
```

## AK/SK-based Authentication

After installing the ModelArts SDK on the local PC, you can perform session authentication using the AK/SK. The sample code is as follows:

```
from modelarts.session import Session

Hardcoded or plaintext AK/SK is risky. For security, encrypt your AK/SK and store them in the
configuration file or environment variables.
In this example, the AK/SK are stored in environment variables for identity authentication. Before running
this example, set environment variables HUAWEICLOUD_SDK_AK and HUAWEICLOUD_SDK_SK.
__AK = os.environ["HUAWEICLOUD_SDK_AK"]
__SK = os.environ["HUAWEICLOUD_SDK_SK"]
Decrypt the password if it is encrypted.
session = Session(access_key=__AK,secret_key=__SK, project_id='****', region_name='****')
```

Parameters are as follows:

- For details about how to obtain the values of **access\_key** and **secret\_key**, see "Obtaining an Access Key" in *User Guide*.
- **project\_id** indicates the project ID. To obtain the project ID, do as follows:
  - a. Log in to the management console.
  - b. In the upper right corner, click your account avatar icon and choose **My Settings** from the drop-down list.
  - c. On the **My Settings** page, go to the **Project List** tab page, which is displayed by default. View the project ID and name in the project list.
- **region\_name** indicates the region ID. Contact the administrator to obtain a region ID.

## 5.2 Authentication Using the Username and Password

This authentication method is available for [OBS Management](#), [Model Management](#), and [Service Management](#).

### Sample Code

- Authentication using an account  
Set **username** to your account name.

```
from modelarts.session import Session
session = Session(username='****', password='****', region_name='****', project_id='****')
```

## 5.3 AK/SK-based Authentication

This authentication method is available for [OBS Management](#), [Model Management](#), and [Service Management](#).

### Sample Code

```
from modelarts.session import Session
session = Session(access_key='****',secret_key='****', project_id='****', region_name='****')
```

Parameters in this command are described as follows:

- For details about how to obtain the values of **access\_key** and **secret\_key**, see "Obtaining an Access Key" in *User Guide*.

- **project\_id** indicates the project ID. To obtain the project ID, do as follows:
  - a. Log in to the management console.
  - b. In the upper right corner, click your account avatar icon and choose **My Settings** from the drop-down list.
  - c. On the **My Settings** page, go to the **Project List** tab page, which is displayed by default. View the project ID and name in the project list.
- **region\_name** indicates the region ID. Contact the administrator to obtain a region ID.

# 6 OBS Management

## 6.1 Overview of OBS Management

ModelArts SDK 1.1.3 supports OBS management, including uploading and downloading files and folders. The operations are as follows:

- [Uploading a File to OBS](#)
- [Uploading a Folder to OBS](#)
- [Downloading a File from OBS](#)
- [Downloading a Folder from OBS](#)

## 6.2 Transferring Files (Recommended)

### NOTE

Through file transferring, local files and folders can be uploaded to OBS, and the files and folders in OBS can be downloaded to a local path.

### Sample Code

In ModelArts notebook, you do not need to enter authentication parameters for session authentication. For details about session authentication of other development environments, see [Session Authentication](#).

```
from modelarts.session import Session
session = Session()
1. Upload a local file to OBS.
session.obs.copy(src_path='/home/ma-user/file1.txt', dst_path='obs://bucket-name/dir1/file1.txt')

2. Download a file from OBS to a local path.
session.obs.copy(src_path='obs://bucket-name/dir1/file1.txt', dst_path='/home/ma-user/file1.txt')

3. Upload a local folder to OBS.
session.obs.copy(src_path='/home/ma-user', dst_path='obs://bucket-name/dir1', keep_last_dir=True)

4. Download a folder from OBS to a local path.
session.obs.copy(src_path='obs://bucket-name/dir1', dst_path='/home/ma-user', keep_last_dir=True)
```



**Table 6-1** Request parameters

| Parameter     | Mandatory | Type    | Description                                                                                                                                                                   |
|---------------|-----------|---------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| session       | Yes       | Object  | Session object.                                                                                                                                                               |
| src_path      | Yes       | String  | Path to the source file or folder. If the source path is an OBS path, the path prefix must be <b>obs://</b> .                                                                 |
| dst_path      | Yes       | String  | Path to the destination file or folder. If the destination path is an OBS path, the path prefix must be <b>obs://</b> .                                                       |
| keep_last_dir | No        | Boolean | Whether to copy the last-level directory of the source folder to the destination folder. The default value is <b>True</b> . This parameter is valid only for copying folders. |

**Table 6-2** Failure parameters

| Parameter     | Type   | Description                                                                                     |
|---------------|--------|-------------------------------------------------------------------------------------------------|
| error_code    | String | Error code when calling the SDK failed. This parameter is unavailable for a successful call.    |
| error_message | String | Error message when calling the SDK failed. This parameter is unavailable for a successful call. |

## 6.3 Uploading a File to OBS

### Sample Code

In ModelArts notebook, you do not need to enter authentication parameters for session authentication. For details about session authentication of other development environments, see [Session Authentication](#).

```
from modelarts.session import Session
session = Session()
session.obs.upload_file(src_local_file='/home/ma-user/file1.txt', dst_obs_dir='obs://bucket-name/dir1/')
```

After the sample code is executed, the local source file **file1.txt** is uploaded to the **dir1** folder in the **bucket-name** bucket. The path is **obs://bucket-name/dir1/file1.txt**. The bucket name and folder name are user-defined.

## Parameters

**Table 6-3** Request parameters

| Parameter      | Mandatory | Type   | Description                                                                                     |
|----------------|-----------|--------|-------------------------------------------------------------------------------------------------|
| session        | Yes       | Object | Session object                                                                                  |
| src_local_file | Yes       | String | Path to the local file to be uploaded                                                           |
| dst_obs_dir    | Yes       | String | Path to the target OBS bucket. The path must start with <b>obs://</b> and end with a slash (/). |

**Table 6-4** Failed response parameters

| Parameter  | Type   | Description                                                                                          |
|------------|--------|------------------------------------------------------------------------------------------------------|
| error_code | String | Error code when the API call fails.<br>This parameter is not included when the API call succeeds.    |
| error_msg  | String | Error message when the API call fails.<br>This parameter is not included when the API call succeeds. |

## 6.4 Uploading a Folder to OBS

### Sample Code

In ModelArts notebook, you do not need to enter authentication parameters for session authentication. For details about session authentication of other development environments, see [Session Authentication](#).

```
from modelarts.session import Session
session = Session()
session.obs.upload_dir(src_local_dir='/home/ma-user/', dst_obs_dir='obs://bucket-name/dir1/')
```

After the sample code is executed, the local source folder **/ma-user/** is uploaded to the **dir1** folder in the **bucket-name** bucket. The path is **obs://bucket-name/dir1/ma-user/**. The bucket name and folder name are user-defined.

## Parameters

**Table 6-5** Request parameters

| Parameter     | Mandatory | Type   | Description                                                                                                                                                                    |
|---------------|-----------|--------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| session       | Yes       | Object | Session object                                                                                                                                                                 |
| src_local_dir | Yes       | String | Path to the local folder to be uploaded. If the folder to be uploaded is empty or contains multiple empty folders, no empty folders are created in the corresponding OBS path. |
| dst_obs_dir   | Yes       | String | Path to the target OBS bucket. The path must start with <b>obs://</b> and end with a slash (/).                                                                                |

**Table 6-6** Failed response parameters

| Parameter  | Type   | Description                                                                                       |
|------------|--------|---------------------------------------------------------------------------------------------------|
| error_code | String | Error code when the API call fails. This parameter is not included when the API call succeeds.    |
| error_msg  | String | Error message when the API call fails. This parameter is not included when the API call succeeds. |

## 6.5 Downloading a File from OBS

### Sample Code

In ModelArts notebook, you do not need to enter authentication parameters for session authentication. For details about session authentication of other development environments, see [Session Authentication](#).

```
from modelarts.session import Session
session = Session()
session.obs.download_file(src_obs_file="obs://bucket-name/dir1/file1.txt", dst_local_dir="/home/ma-user/")
```

After the sample code is executed, source file **file1.txt** is downloaded from OBS to **/home/ma-user/file1.txt**.

## Parameters

**Table 6-7** Request parameters

| Parameter     | Mandatory | Type   | Description                                                                                 |
|---------------|-----------|--------|---------------------------------------------------------------------------------------------|
| session       | Yes       | Object | Session object                                                                              |
| src_obs_file  | Yes       | String | Path to the source file to be downloaded from OBS. The path must start with <b>obs://</b> . |
| dst_local_dir | Yes       | String | Path to the target local folder. The path must end with a slash (/).                        |

**Table 6-8** Failed response parameters

| Parameter  | Type   | Description                                                                                       |
|------------|--------|---------------------------------------------------------------------------------------------------|
| error_code | String | Error code when the API call fails. This parameter is not included when the API call succeeds.    |
| error_msg  | String | Error message when the API call fails. This parameter is not included when the API call succeeds. |

## 6.6 Downloading a Folder from OBS

### Sample Code

In ModelArts notebook, you do not need to enter authentication parameters for session authentication. For details about session authentication of other development environments, see [Session Authentication](#).

```
from modelarts.session import Session
session = Session()
session.obs.download_dir(src_obs_dir="obs://bucket-name/dir1/", dst_local_dir="/home/ma-user/work/")
```

After the sample code is executed, source folder **dir1** is downloaded from OBS to **/home/ma-user/work/dir1/**.

#### CAUTION

You must have the write permission on the local path.

## Parameters

**Table 6-9** Request parameters

| Parameter     | Mandatory | Type   | Description                                                                                                                                                                                                                          |
|---------------|-----------|--------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| session       | Yes       | Object | Session object                                                                                                                                                                                                                       |
| src_obs_dir   | Yes       | String | Path to the source folder to be downloaded from OBS. The path must start with <b>obs://</b> and end with a slash (/). If the downloaded folder contains empty folders, no empty folders are created in the corresponding local path. |
| dst_local_dir | Yes       | String | Path to the target local folder. The path must end with a slash (/).                                                                                                                                                                 |

**Table 6-10** Failed response parameters

| Parameter  | Type   | Description                                                                                       |
|------------|--------|---------------------------------------------------------------------------------------------------|
| error_code | String | Error code when the API call fails. This parameter is not included when the API call succeeds.    |
| error_msg  | String | Error message when the API call fails. This parameter is not included when the API call succeeds. |

# 7 Data Management

## 7.1 Managing Datasets

### 7.1.1 Querying a Dataset List

Obtain a dataset list by page.

```
list_datasets(session, dataset_type=None, dataset_name=None, offset=None, limit=None)
```

#### Sample Code

- Example 1: Obtain a dataset list.

```
from modelarts.session import Session
from modelarts.dataset import Dataset
session = Session()
Obtain a dataset list.
dataset_list = Dataset.list_datasets(session)
print(dataset_list) # Print the query result.
```
- Example 2: Obtain a dataset list by dataset type.

```
Obtain image classification datasets.
dataset_list = Dataset.list_datasets(session, dataset_type=0)
print(dataset_list)
```
- Example 3: Obtain a dataset list by dataset name.

```
Obtain the datasets with dataset contained in dataset names.
dataset_list = Dataset.list_datasets(session, dataset_name="dataset")
print(dataset_list)
```
- Example 4: Obtain a dataset list by page.

```
By default, 10 dataset records are returned at a time. You can set limit and offset for query by page.
dataset_list = Dataset.list_datasets(session, offset=0, limit=50) # Obtain the 1st to 50th records.
print(dataset_list)
dataset_list = Dataset.list_datasets(session, offset=1, limit=50) # Obtain the 51st to 100th records.
print(dataset_list)
```

## Parameters

**Table 7-1** Request parameters

| Name         | Mandatory | Type    | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |
|--------------|-----------|---------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| session      | Yes       | Object  | Session object. For details about the initialization method, see <a href="#">Session Authentication</a> .                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
| dataset_type | No        | Integer | Obtain a dataset list by dataset type. By default, this parameter is left blank. The options are as follows: <ul style="list-style-type: none"> <li>• <b>0</b>: image classification</li> <li>• <b>1</b>: object detection</li> <li>• <b>3</b>: image segmentation</li> <li>• <b>100</b>: text classification</li> <li>• <b>101</b>: named entity recognition</li> <li>• <b>102</b>: text triplet</li> <li>• <b>200</b>: sound classification</li> <li>• <b>201</b>: speech content</li> <li>• <b>202</b>: speech paragraph labeling</li> <li>• <b>400</b>: table dataset</li> <li>• <b>600</b>: video labeling</li> <li>• <b>900</b>: custom format</li> </ul> |
| dataset_name | No        | String  | Fuzzy search keyword. By default, this parameter is left blank.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |
| offset       | No        | Integer | Start page for pagination display. The default value is <b>0</b> .                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
| limit        | No        | Integer | Maximum number of records returned on each page. The value ranges from 1 to 100. The default value is <b>10</b> .                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |

### 7.1.2 Creating a Dataset

Create a dataset whose data can be imported from OBS.

```
create_dataset(session, dataset_name=None, data_type=None, data_sources=None, work_path=None, dataset_type=None, **kwargs)
```

Use either of the following methods to create a dataset:

- Create a dataset based on the labeling type. One dataset supports only one labeling task type.

```
create_dataset(session,dataset_name=None, dataset_type=None, data_sources=None,
work_path=None, **kwargs)
```

- Create a dataset based on the data type. You can create different types of labeling tasks on the same dataset. For example, create image classification and object detection labeling tasks on an image dataset.

```
create_dataset(session,dataset_name=None, data_type=None, data_sources=None, work_path=None,
**kwargs)
```

#### NOTE

You are advised to create a dataset based on the data type. Creating a dataset based on the labeling type will be terminated.

## Sample Code

- Example 1: Create an image dataset based on the data type.

```
from modelarts.session import Session
from modelarts.dataset import Dataset

session = Session()

dataset_name = "dataset-image" # Dataset name
data_type = "IMAGE" # Dataset type, which is an image dataset
data_sources = dict() # Dataset data source
data_sources["type"] = 0 # Data source type. Value 0 indicates OBS.
data_sources["path"] = "/obs-gaia-test/data/image/image-classification/" # Path for storing data in
OBS
work_path = dict() # Work directory of the dataset
work_path["type"] = 0 # Working directory type of the dataset. Value 0 indicates OBS.
work_path["path"] = "/obs-gaia-test/data/output/work_path/" # Path for the working directory of the
dataset in OBS
create_dataset_resp = Dataset.create_dataset(session, dataset_name=dataset_name,
data_type=data_type,
data_sources=data_sources, work_path=work_path)
```

- Example 2: Create an image dataset based on the data types (labels imported).

```
from modelarts.session import Session
from modelarts.dataset import Dataset

session = Session()

dataset_name = "dataset-image-with-annotations"
data_type = "IMAGE"
data_sources = dict()
data_sources["type"] = 0
data_sources["path"] = "/obs-gaia-test/data/image/image-classification/"
annotation_config = dict() # Labeling format of the source data
annotation_config["scene"] = "image_classification" # Image classification labeling
annotation_config["format_name"] = "ModelArts image classification 1.0" # Labeling format of
ModelArts image classification 1.0
data_sources["annotation_config"] = annotation_config
work_path = dict()
work_path["type"] = 0
work_path["path"] = "/obs-gaia-test/data/output/work_path/"
create_dataset_resp = Dataset.create_dataset(session, dataset_name=dataset_name,
data_type=data_type,
data_sources=data_sources, work_path=work_path)
```

- Example 3: Create a table dataset based on the data type.

```
from modelarts.session import Session
from modelarts.dataset import Dataset

session = Session()

dataset_name = "dataset-table"
data_type = "TABLE"
data_sources = dict()
```



```
data_sources["type"] = 0
data_sources["path"] = "/obs-gaia-test/data/table/table0/"
data_sources["with_column_header"] = True
work_path = dict()
work_path["type"] = 0
work_path["path"] = "/obs-gaia-test/data/output/work_path/"
Schema information of the table data needs to be specified for the table dataset.
schema0 = dict()
schema0["schema_id"] = 0
schema0["name"] = "name"
schema0["type"] = "STRING"
schema1 = dict()
schema1["schema_id"] = 1
schema1["name"] = "age"
schema1["type"] = "STRING"
schema2 = dict()
schema2["schema_id"] = 2
schema2["name"] = "label"
schema2["type"] = "STRING"
schemas = []
schemas.append(schema0)
schemas.append(schema1)
schemas.append(schema2)
create_dataset_resp = Dataset.create_dataset(session, dataset_name=dataset_name,
data_type=data_type,
data_sources=data_sources, work_path=work_path, schema=schemas)
```

- **Example 4: Create an image classification dataset based on the labeling type.**

```
from modelarts.session import Session
from modelarts.dataset import Dataset

session = Session()

dataset_name = "dataset-image-classification"
dataset_type = 0 # Dataset labeling type. Value 0 indicates image classification.
data_sources = dict()
data_sources["path"] = "/obs-gaia-test/data/image/image-classification/"
data_sources["type"] = "0"
work_path = dict()
work_path["type"] = 0
work_path["path"] = "/obs-gaia-test/data/output/work_path/"
create_dataset_resp = Dataset.create_dataset(session, dataset_name=dataset_name,
dataset_type=dataset_type, data_sources=data_sources, work_path=work_path)
```

- **Example 5: Create a text triplet dataset based on the labeling type.**

```
dataset_name = "dataset-text-triplet"
dataset_type = 102 # Dataset labeling type. Value 102 indicates text triplet.
data_sources = dict()
data_sources["type"] = 0
data_sources["path"] = "/obs-gaia-test/data/text/text-classification/"
work_path = dict()
work_path["type"] = 0
work_path["path"] = "/obs-gaia-test/data/output/work_path/"

Create a dataset of the text triplet labeling type with labels imported.
label_entity1 = dict() # Label object
label_entity1["name"] = "Disease" # Label name
label_entity1["type"] = 101 # Label type. Value 101 indicates an entity.
label_entity2 = dict()
label_entity2["name"] = "Disease alias"
label_entity2["type"] = 101
label_relation1 = dict()
label_relation1["name"] = "Also called"
label_relation1["type"] = 102 # Label type. Value 102 indicates relational.
property = dict() # For a relational label, the start entity label and end entity label must be
specified in label properties.
property["@modelarts:from_type"] = "Disease" # Start entity label
property["@modelarts:to_type"] = "Disease alias" # End entity label
label_relation1["property"] = property
labels = []
```

```
labels.append(label_entity1)
labels.append(label_entity2)
labels.append(label_relation1)
create_dataset_resp = Dataset.create_dataset(session, dataset_name=dataset_name,
dataset_type=dataset_type, data_sources=data_sources, work_path=work_path, labels=labels)
```

- Example 6: Create a table dataset based on the labeling type.

```
dataset_name = "dataset-table"
dataset_type = 400 # Dataset labeling type. Value 400 indicates a table dataset.
data_sources = dict()
data_sources['type'] = 0
data_sources['path'] = "/obs-gaia-test/data/table/table0/"
data_sources['with_column_header'] = True # Whether the table data contains a table header
work_path = dict()
work_path['type'] = 0
work_path['path'] = "/obs-gaia-test/data/output/work_path/"

The table header of the table data needs to be imported to the table dataset.
schema0 = dict() # Table header
schema0['schema_id'] = 0 # Header of the first column
schema0['name'] = "name" # Table header name, which is name in the column
schema0['type'] = "STRING" # Data type of the table header, indicating a character string
schema1 = dict()
schema1['schema_id'] = 1
schema1['name'] = "age"
schema1['type'] = "STRING"
schema2 = dict()
schema2['schema_id'] = 2
schema2['name'] = "label"
schema2['type'] = "STRING"
schemas = []
schemas.append(schema0)
schemas.append(schema1)
schemas.append(schema2)
create_dataset_resp = Dataset.create_dataset(session, dataset_name=dataset_name,
dataset_type=dataset_type, data_sources=data_sources, work_path=work_path, schema=schemas)
```

## Parameters

**Table 7-2** Request parameters

| Name         | Mandatory | Type   | Description                                                                                               |
|--------------|-----------|--------|-----------------------------------------------------------------------------------------------------------|
| session      | Yes       | Object | Session object. For details about the initialization method, see <a href="#">Session Authentication</a> . |
| dataset_name | Yes       | String | Dataset name                                                                                              |

| Name         | Mandatory | Type    | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
|--------------|-----------|---------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| data_type    | No        | String  | <p>Data type of a dataset. Either <b>data_type</b> or <b>dataset_type</b> must be specified. <b>data_type</b> is recommended. The options are as follows:</p> <ul style="list-style-type: none"> <li>● <b>IMAGE</b>: image</li> <li>● <b>TEXT</b>: text</li> <li>● <b>AUDIO</b>: audio</li> <li>● <b>TABLE</b>: table</li> <li>● <b>VIDEO</b>: video</li> <li>● <b>PLAIN</b>: custom format</li> </ul>                                                                                                                                                                                                                                                                                                   |
| dataset_type | No        | Integer | <p>Obtain a dataset list based on the dataset type. Either <b>data_type</b> or <b>dataset_type</b> must be specified. The options are as follows:</p> <ul style="list-style-type: none"> <li>● <b>0</b>: image classification</li> <li>● <b>1</b>: object detection</li> <li>● <b>3</b>: image segmentation</li> <li>● <b>100</b>: text classification</li> <li>● <b>101</b>: named entity recognition</li> <li>● <b>102</b>: text triplet</li> <li>● <b>200</b>: sound classification</li> <li>● <b>201</b>: speech content</li> <li>● <b>202</b>: speech paragraph labeling</li> <li>● <b>400</b>: table dataset</li> <li>● <b>600</b>: video labeling</li> <li>● <b>900</b>: custom format</li> </ul> |

| Name         | Mandatory | Type                               | Description                                                                                                                                                                                                                                                                                                                   |
|--------------|-----------|------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| data_sources | Yes       | <a href="#">Table 7-3</a>          | Input dataset path, which is used to synchronize source data (such as images, text files, and audio files) in the directory and its subdirectories to the dataset. For a table dataset, this parameter indicates the import directory. The work directory of a table dataset cannot be an OBS path in a KMS-encrypted bucket. |
| work_path    | Yes       | <a href="#">Table 7-7</a>          | Output dataset path, which is used to store output files such as label files.                                                                                                                                                                                                                                                 |
| labels       | No        | List of <a href="#">Table 7-8</a>  | Dataset labels. This parameter must be imported when you create a text triplet dataset.                                                                                                                                                                                                                                       |
| schema       | No        | List of <a href="#">Table 7-10</a> | Schema list, which is used to specify the name and type of the table header of a table dataset                                                                                                                                                                                                                                |
| description  | No        | String                             | Dataset description consisting of 0 to 256 characters without special characters (^!<>=&"'). The parameter is left blank by default.                                                                                                                                                                                          |

**Table 7-3 DataSource** parameters

| Name | Mandatory | Type    | Description                                                                                                                                                                             |
|------|-----------|---------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| type | Yes       | Integer | Data type. The options are as follows: <ul style="list-style-type: none"> <li>• <b>0</b>: OBS bucket (default value)</li> <li>• <b>5</b>: Dataset downloaded from AI Gallery</li> </ul> |

| Name               | Mandatory | Type                      | Description                                                                                                                                                                                                                                                                                                                          |
|--------------------|-----------|---------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| path               | Yes       | String                    | Data source path <ul style="list-style-type: none"> <li>Newline characters (\n), carriage return characters (\r), and tab characters (\t) are not allowed.</li> </ul>                                                                                                                                                                |
| content_info       | No        | <a href="#">Table 7-4</a> | Dataset asset downloaded from the AI Gallery                                                                                                                                                                                                                                                                                         |
| annotation_config  | No        | <a href="#">Table 7-5</a> | Data labeling format, which can be: <ul style="list-style-type: none"> <li>Image classification</li> <li>Object detection</li> <li>Text classification</li> <li>Sound classification</li> </ul>                                                                                                                                      |
| with_column_header | No        | Boolean                   | Whether the first row of a table is the table header. This parameter is mandatory for table datasets. <ul style="list-style-type: none"> <li><b>True:</b> The first row of a table is used as the table header.</li> <li><b>False:</b> The first row of a table is not used as the table header, but only as sample data.</li> </ul> |

**Table 7-4 ContentInfo** parameters

| Name       | Mandatory | Type   | Description                            |
|------------|-----------|--------|----------------------------------------|
| content_id | Yes       | String | Dataset asset ID in AI Gallery         |
| version_id | Yes       | String | Dataset asset version ID in AI Gallery |

**Table 7-5 AnnotationConfig** parameters

| Name        | Mandatory | Type                      | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |
|-------------|-----------|---------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| scene       | Yes       | String                    | Supported labeling scenarios. The options are as follows: <ul style="list-style-type: none"> <li>• <b>image_classification</b></li> <li>• <b>object_detection</b></li> <li>• <b>text_classification</b></li> <li>• <b>audio_classification</b></li> </ul>                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |
| format_name | Yes       | String                    | Labeling format in different scenarios. The options are as follows: <ul style="list-style-type: none"> <li>• <b>image_classification</b> <ul style="list-style-type: none"> <li>- ModelArts imageNet 1.0</li> <li>- ModelArts image classification 1.0</li> </ul> </li> <li>• <b>object_detection</b> <ul style="list-style-type: none"> <li>- ModelArts PASCAL VOC 1.0</li> <li>- YOLO</li> </ul> </li> <li>• <b>text_classification</b> <ul style="list-style-type: none"> <li>- ModelArts text classification 1.0</li> <li>- ModelArts text classification combine 1.0</li> </ul> </li> <li>• <b>audio_classification</b> <ul style="list-style-type: none"> <li>- ModelArts audio classification dir 1.0</li> </ul> </li> </ul> |
| parameters  | No        | <a href="#">Table 7-6</a> | Advanced labeling format parameters, such as the sample separator                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |

**Table 7-6 AnnotationConfigParam** parameters

| Name            | Mandatory | Type                              | Description                                |
|-----------------|-----------|-----------------------------------|--------------------------------------------|
| included_labels | No        | List of <a href="#">Table 7-8</a> | Import only samples with specified labels. |

| Name                   | Mandatory | Type    | Description                                                                                                                                                                                        |
|------------------------|-----------|---------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| sample_label_separator | No        | String  | Separator between text and labels. The separator contains only one character, which must be a letter, digit, or one of the following characters (@#¥%^&*_*=?/!':;). The separator must be escaped. |
| label_separator        | No        | String  | Separator between labels. The separator contains only one character, which must be a letter, digit, or one of the following characters (@#¥%^&*_*=?/!':;). The separator must be escaped.          |
| difficult_only         | No        | Boolean | Whether to import only hard examples.                                                                                                                                                              |

**Table 7-7 WorkPath** parameters

| Parameter | Mandatory | Type    | Description                                                                                                                     |
|-----------|-----------|---------|---------------------------------------------------------------------------------------------------------------------------------|
| type      | Yes       | Integer | Data type. The options are as follows: <ul style="list-style-type: none"> <li>• <b>0</b>: OBS bucket (default value)</li> </ul> |

| Parameter | Mandatory | Type   | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
|-----------|-----------|--------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| path      | Yes       | String | <p>Output dataset path, which is used to store output files such as label files.</p> <ul style="list-style-type: none"> <li>• The format is "/Bucket name/File path", for example, <b>/obs-bucket/flower/rose/</b> (directory used as the path).</li> <li>• A bucket cannot be used as a path.</li> <li>• The output path must be different from the input path and its subdirectories.</li> <li>• The parameter consists of 3 to 700 characters.</li> <li>• Newline characters (\n), carriage return characters (\r), and tab characters (\t) are not allowed.</li> </ul> |

**Table 7-8** Label parameters

| Parameter | Mandatory | Type                      | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
|-----------|-----------|---------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| name      | Yes       | String                    | Label name                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |
| type      | Yes       | Integer                   | <p>Label type. The options are as follows:</p> <ul style="list-style-type: none"> <li>• <b>0</b>: image classification</li> <li>• <b>1</b>: object detection</li> <li>• <b>3</b>: image segmentation</li> <li>• <b>100</b>: text classification</li> <li>• <b>101</b>: named entity</li> <li>• <b>102</b>: text triplet relationship</li> <li>• <b>200</b>: sound classification</li> <li>• <b>201</b>: speech content</li> <li>• <b>202</b>: speech paragraph labeling</li> <li>• <b>600</b>: video labeling</li> </ul> |
| property  | No        | <a href="#">Table 7-9</a> | Basic attribute key-value pair of a label, such as color                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |



**Table 7-9 LabelProperty parameters**

| Parameter            | Mandatory | Type   | Description                                                                                                                                                                                                |
|----------------------|-----------|--------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| @modelarts:color     | No        | String | (Built-in attribute) Label color, which is a hexadecimal code of the color. By default, this parameter is left blank. For example, #FFFFFF0.                                                               |
| @modelarts:from_type | No        | String | (Built-in attribute) Type of the head entity in a triplet relationship label. This attribute must be specified when a relationship label is created. This parameter is only used in text triplet datasets. |
| @modelarts:to_type   | No        | String | (Built-in attribute) Type of the tail entity in a triplet relationship label. This attribute must be specified when a relationship label is created. This parameter is only used in text triplet datasets. |

**Table 7-10 Schema parameters**

| Parameter | Mandatory | Type    | Description |
|-----------|-----------|---------|-------------|
| schema_id | No        | Integer | Schema ID   |
| name      | No        | String  | Schema name |

| Parameter   | Mandatory | Type   | Description                                                                                                                                                                                                                                                          |
|-------------|-----------|--------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| type        | No        | String | Schema value type. The options are as follows: <ul style="list-style-type: none"> <li>• STRING</li> <li>• SHORT</li> <li>• INT</li> <li>• LONG</li> <li>• DOUBLE</li> <li>• FLOAT</li> <li>• BYTE</li> <li>• DATE</li> <li>• TIMESTAMP</li> <li>• BOOLEAN</li> </ul> |
| description | No        | String | Schema description                                                                                                                                                                                                                                                   |

### 7.1.3 Querying Details About a Dataset

Obtain details about a dataset, including the samples and versions of the dataset.

```
dataset.get_dataset_info()
```

#### Sample Code

Obtain details about a dataset.

```
from modelarts.session import Session
from modelarts.dataset import Dataset
session = Session()

dataset = Dataset(session, dataset_id)
dataset_info = dataset.get_dataset_info()
print(dataset_info) # Output the detailed information about the dataset.
```

#### Parameters

None

### 7.1.4 Modifying a Dataset

Change the name or modify the description of a dataset.

```
dataset.update_dataset(dataset_name=None, description=None)
```

#### Sample Code

Change a dataset name.

```
from modelarts.session import Session
from modelarts.dataset import Dataset
session = Session()
```

```
dataset = Dataset(session, dataset_id)
dataset.update_dataset(dataset_name = "new-dataset-name")
```

## Parameters

**Table 7-11** Request parameters

| Parameter    | Mandatory | Type   | Description         |
|--------------|-----------|--------|---------------------|
| dataset_name | No        | String | Dataset name        |
| description  | No        | String | Dataset description |

## 7.1.5 Deleting a Dataset

Delete a dataset based on the dataset ID.

```
delete_dataset(session, dataset_id)
```

### Sample Code

Delete a dataset.

```
from modelarts.session import Session
from modelarts.dataset import Dataset
session = Session()

Dataset.delete_dataset(session, dataset_id="68ZXdk6CZwgvUICOOdC")
```

## Parameters

**Table 7-12** Request parameters

| Parameter  | Mandatory | Type   | Description                                                                                               |
|------------|-----------|--------|-----------------------------------------------------------------------------------------------------------|
| session    | Yes       | Object | Session object. For details about the initialization method, see <a href="#">Session Authentication</a> . |
| dataset_id | Yes       | String | Dataset ID                                                                                                |

## 7.2 Managing Dataset Versions

### 7.2.1 Obtaining a Dataset Version List

Obtain a list of dataset versions.

```
dataset.list_versions()
```

### Sample Code

Obtain a dataset version list.

```
from modelarts.session import Session
from modelarts.dataset import Dataset
session = Session()

dataset = Dataset(session, dataset_id)
version_list = dataset.list_versions()
print(version_list) # Print the dataset version list.
```

## Parameters

None

## 7.2.2 Creating a Dataset Version

Create a new version for a dataset.

```
dataset.create_version(name=None, version_format=None, label_task_type=None, label_task_id=None,
**kwargs)
```

## Sample Code

Example 1: Create a new version for a dataset.

```
from modelarts.session import Session
from modelarts.dataset import Dataset
session = Session()

dataset = Dataset(session, dataset_id)
create_version_resp = dataset.create_version(name="V001", version_format="Default", label_task_type=0,
description="version 001")
```

Example 2: Create a dataset based on a labeling task.

```
from modelarts.session import Session
from modelarts.dataset import Dataset
session = Session()

dataset = Dataset(session, dataset_id)
create_version_resp = dataset.create_version(label_task_id="IbAhFai5KXWC3gthUfz", description="dataset
version from label task")
```

## Parameters

**Table 7-13** Request parameters

| Parameter      | Mandatory | Type   | Description                                                                                                               |
|----------------|-----------|--------|---------------------------------------------------------------------------------------------------------------------------|
| name           | No        | String | Version name that consists of 1 to 32 characters.                                                                         |
| version_format | No        | String | Format of a dataset version. The options are as follows: <ul style="list-style-type: none"> <li><b>Default</b></li> </ul> |

| Parameter       | Mandatory | Type    | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |
|-----------------|-----------|---------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| label_task_type | No        | Integer | Labeling type of a dataset version. The options are as follows: <ul style="list-style-type: none"> <li>• <b>0</b>: image classification</li> <li>• <b>1</b>: object detection</li> <li>• <b>3</b>: image segmentation</li> <li>• <b>100</b>: text classification</li> <li>• <b>101</b>: named entity recognition</li> <li>• <b>102</b>: text triplet</li> <li>• <b>200</b>: sound classification</li> <li>• <b>201</b>: speech content</li> <li>• <b>202</b>: speech paragraph labeling</li> <li>• <b>400</b>: table dataset</li> <li>• <b>600</b>: video labeling</li> <li>• <b>900</b>: custom format</li> </ul> |
| label_task_id   | No        | String  | ID of a labeling task based on which a dataset version is created.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |
| description     | No        | String  | Version description consisting of 0 to 256 characters without special characters (!<=&"'). The parameter is left blank by default.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |

### 7.2.3 Querying Details About a Dataset Version

Obtain details about a dataset version based on the version ID.

```
dataset.get_version_info(version_id)
```

#### Sample Code

Obtain details about a dataset version.

```
from modelarts.session import Session
from modelarts.dataset import Dataset
session = Session()

dataset = Dataset(session, dataset_id)
version_info = dataset.get_version_info(version_id="cSPuXPgnYp7ObRs6LaR")
print(version_info) # Print details about the dataset version.
```

## Parameters

**Table 7-14** Request parameters

| Parameter  | Mandatory | Type   | Description        |
|------------|-----------|--------|--------------------|
| version_id | Yes       | String | Dataset version ID |

### 7.2.4 Deleting a Dataset Version

Delete a specified dataset version.

```
dataset.delete_version(version_id)
```

#### Sample Code

Delete a specified dataset version.

```
from modelarts.session import Session
from modelarts.dataset import Dataset
session = Session()

dataset = Dataset(session, dataset_id)
dataset.delete_version(version_id="cSPuXPgnYp7ObRs6LaR")
```

## Parameters

**Table 7-15** Request parameters

| Parameter  | Mandatory | Type   | Description        |
|------------|-----------|--------|--------------------|
| version_id | Yes       | String | Dataset version ID |

## 7.3 Managing Samples

### 7.3.1 Querying a Sample List

Obtain the sample list of a dataset. Table datasets are not supported.

```
dataset.list_samples(version_id=None, offset=None, limit=None)
```

#### Sample Code

- Example 1: Obtain a dataset sample list.

```
from modelarts.session import Session
from modelarts.dataset import Dataset
session = Session()

dataset = Dataset(session, dataset_id)
list_samples_resp = dataset.list_samples()
print(list_samples_resp) # Print the sample list.
```

- Example 2: Obtain the sample list of a specified dataset version.  

```
list_samples_resp = dataset.list_samples(version_id = "cSPuXPgnYp7ObRs6LaR")
print(list_samples_resp)
```

## Parameters

**Table 7-16** Request parameters

| Parameter  | Mandatory | Type    | Description                                                                                                       |
|------------|-----------|---------|-------------------------------------------------------------------------------------------------------------------|
| version_id | No        | String  | Dataset version ID, which can be used for obtaining the sample list of this dataset version.                      |
| offset     | No        | Integer | Start page for pagination display. The default value is <b>0</b> .                                                |
| limit      | No        | Integer | Maximum number of records returned on each page. The value ranges from 1 to 100. The default value is <b>10</b> . |

### 7.3.2 Querying Details About a Sample

Obtain details about a specified sample in a dataset based on the sample ID.

```
dataset.get_sample_info(sample_id)
```

## Sample Code

Obtain details about a specified sample in a dataset based on the sample ID.

```
from modelarts.session import Session
from modelarts.dataset import Dataset
session = Session()

dataset = Dataset(session, dataset_id)
sample_info = dataset.get_sample_info(sample_id="2551e78974aed9b60156d8376232f6bd")
print(sample_info) # Print the detailed information about the sample.
```

## Parameters

**Table 7-17** Request parameters

| Parameter | Mandatory | Type   | Description |
|-----------|-----------|--------|-------------|
| sample_id | Yes       | String | Sample ID   |

### 7.3.3 Deleting Samples in a Batch

Delete samples from a dataset in a batch based on the sample ID list.

```
dataset.delete_samples(samples)
```

## Sample Code

Delete samples from a dataset in a batch.

```
from modelarts.session import Session
from modelarts.dataset import Dataset
session = Session()

dataset = Dataset(session, dataset_id)
samples = []
samples.append("2551e78974aed9b60156d8376232f6bd")
samples.append("0d315fec1efc7568de5cccf522c10a1b")
dataset.delete_samples(samples)
```

## Parameters

**Table 7-18** Request parameters

| Parameter | Mandatory | Type           | Description                      |
|-----------|-----------|----------------|----------------------------------|
| samples   | Yes       | List of String | IDs of the samples to be deleted |

## 7.4 Managing Dataset Import Tasks

### 7.4.1 Querying a Dataset Import Task List

Obtain a dataset import task list.

```
dataset.list_import_tasks()
```

## Sample Code

Obtain a dataset import task list.

```
from modelarts.session import Session
from modelarts.dataset import Dataset
session = Session()

dataset = Dataset(session, dataset_id)
list_tasks_resp = dataset.list_import_tasks()
print(list_tasks_resp) # Print the import task list.
```

## Parameters

None

### 7.4.2 Creating a Dataset Import Task

You can import new data from OBS through an OBS path or a manifest file.

```
dataset.import_data(path=None, annotation_config=None, **kwargs)
```

**Table 7-19** lists the import modes supported by datasets.



**Table 7-19** Import modes supported by datasets

| Dataset Type              | From an OBS Path | From a Manifest File | Remarks                                                                         |
|---------------------------|------------------|----------------------|---------------------------------------------------------------------------------|
| Image classification      | Supported        | Supported            | None                                                                            |
| Object detection          | Supported        | Supported            | None                                                                            |
| Image segmentation        | Supported        | Supported            | None                                                                            |
| Text classification       | Supported        | Supported            | None                                                                            |
| Named entity recognition  | Not supported    | Supported            | None                                                                            |
| Text triplet              | Not supported    | Supported            | None                                                                            |
| Sound classification      | Supported        | Supported            | None                                                                            |
| Speech labeling           | Not supported    | Supported            | None                                                                            |
| Speech paragraph labeling | Not supported    | Supported            | None                                                                            |
| Table dataset             | Supported        | Not supported        | The schema of the newly imported table data is the same as that of the dataset. |
| Video labeling            | Not supported    | Supported            | None                                                                            |

## Sample Code

- Example 1: Import an object detection dataset from an OBS path.

```

from modelarts.session import Session
from modelarts.dataset import Dataset
session = Session()

dataset = Dataset(session, dataset_id)
annotation_config = dict()
annotation_config['scene'] = "object_detection"
annotation_config['format_name'] = "ModelArts PASCAL VOC 1.0"
import_resp = dataset.import_data(path="/obs-gaia-test/data/image/image-detection/",
annotation_config=annotation_config)

```
- Example 2: Import an object detection dataset from a manifest file.

```

annotation_config = dict() # Task with data imported from a manifest file. annotation_config is
used to import labels.
import_resp = dataset.import_data(
 path="/obs-gaia-test/data/output/work_path/dataset-5932-Qdd1RUZ3wqBQrwrTr3v/
annotation/V001/V001.manifest",annotation_config=annotation_config)

```
- Example 3: Import a table dataset from an OBS path.

```

import_resp = dataset.import_data(
 path="/obs-gaia-test/data/table/table1/", with_column_header=True)

```

## Parameters

**Table 7-20** Request parameters

| Parameter | Mandatory | Type   | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |
|-----------|-----------|--------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| path      | Yes       | String | <p>OBS path or manifest file path for importing data</p> <ul style="list-style-type: none"> <li>• If data is to be imported from a manifest file, ensure the manifest file is specified in the path.</li> <li>• If data is to be imported from an OBS path, ensure only image classification, object detection, image segmentation, text classification, sound classification, and table datasets are supported.</li> <li>• Newline characters (\n), carriage return characters (\r), and tab characters (\t) are not allowed.</li> </ul> |

| Parameter          | Mandatory | Type                      | Description                                                                                                                                                                                                                                                                                                                                                                                                                     |
|--------------------|-----------|---------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| annotation_config  | No        | <a href="#">Table 7-5</a> | <p>Data labeling format. If this parameter is set to <b>None</b>, no labels will be imported. If data is to be imported from a manifest file, import an empty dict object so that labels can be imported. The following labeling formats are supported:</p> <ul style="list-style-type: none"> <li>• Image classification</li> <li>• Object detection</li> <li>• Sound classification</li> <li>• Text classification</li> </ul> |
| with_column_header | No        | Boolean                   | <p>Whether the first row of a table is the table header. This parameter is mandatory for table datasets.</p> <ul style="list-style-type: none"> <li>• <b>True:</b> The first row of a table is used as the table header.</li> <li>• <b>False:</b> The first row of a table is not used as the table header, but only as sample data.</li> </ul>                                                                                 |

### 7.4.3 Querying the Status of a Dataset Import Task

Obtain the status and details of a dataset import task based on the task ID.

```
dataset.get_import_task_info(task_id)
```

## Sample Code

Obtain details about a dataset import task.

```
from modelarts.session import Session
from modelarts.dataset import Dataset
session = Session()

dataset = Dataset(session, dataset_id)
task_info = dataset.get_import_task_info(task_id="r4R52nJ4VJKcivuiocu")
print(task_info) # Print the detailed information about the import task.
```

## Parameters

**Table 7-21** Request parameters

| Parameter | Mandatory | Type   | Description          |
|-----------|-----------|--------|----------------------|
| task_id   | Yes       | String | ID of an import task |

# 7.5 Managing Export Tasks

## 7.5.1 Querying a Dataset Export Task List

Obtain a dataset export task list.

```
dataset.list_export_tasks()
```

## Sample Code

Obtain a dataset export task list.

```
from modelarts.session import Session
from modelarts.dataset import Dataset
session = Session()

dataset = Dataset(session, dataset_id)
list_tasks_resp = dataset.list_export_tasks()
print(list_tasks_resp) # Print the export task list.
```

## Parameters

None

## 7.5.2 Creating a Dataset Export Task

Export the samples of a dataset to a specified OBS path. This function is only supported by image classification, object detection, image segmentation, and custom format datasets.

```
dataset.export_data(path)
```

## Sample Code

Export the samples of a dataset to an OBS path.

```
from modelarts.session import Session
from modelarts.dataset import Dataset
```

```
session = Session()
dataset = Dataset(session, dataset_id)
export_resp = dataset.export_data("/obs-gaia-test/data/output/export-test/")
```

## Parameters

**Table 7-22** Request parameters

| Parameter | Mandatory | Type   | Description                            |
|-----------|-----------|--------|----------------------------------------|
| path      | Yes       | String | OBS path for storing the exported data |

## 7.5.3 Querying the Status of a Dataset Export Task

Obtain the status and details of a dataset export task based on the task ID.

```
dataset.get_export_task_info(task_id)
```

## Sample Code

Obtain the status of a dataset export task.

```
from modelarts.session import Session
from modelarts.dataset import Dataset
session = Session()

dataset = Dataset(session, dataset_id)
task_info = dataset.get_export_task_info(task_id="iuHALF6xdkSAGKVN2jD")
print(task_info) # Print the detailed information about the export task.
```

## Parameters

**Table 7-23** Request parameters

| Parameter | Mandatory | Type   | Description          |
|-----------|-----------|--------|----------------------|
| task_id   | Yes       | String | ID of an export task |

## 7.6 Managing Manifest Files

### 7.6.1 Overview of Manifest Management

When using ModelArts, perform operations such as labeling data, training a model, performing inference, managing datasets, and publishing a model in AI Gallery. All these operations are based on datasets. To standardize the use of datasets in various application scenarios and ensure the flexibility of dataset management, this document describes the manifest file with dataset management APIs and specifications included.

- A manifest file defines the mapping between labeled objects and content. The manifest file can contain only unlabeled data, for example, a created dataset that has not been labeled.
- A manifest file is encoded using UTF-8. Therefore, the programs processing manifest must support UTF-8.
- You can create a manifest file or obtain such a file using a third-party tool or ModelArts.
- Any valid file name is allowed for a manifest file.

## 7.6.2 Parsing a Manifest File

Parse a manifest file in either a local or OBS path. If an OBS path is used, a session is required.

```
manifest.parse_manifest(manifest_path, encoding='utf-8')
```

### Sample Code

```
Parse a manifest file.
from modelarts.session import Session
from modelarts.dataset.format.manifest import Manifest

path = "obs://your-obs-bucket/manifest/V001.manifest"
session = Session()
manifest_info= Manifest.parse_manifest(path,session=session)
```

### Parameters

**Table 7-24** Request parameters

| Parameter     | Mandatory | Type   | Description                                                                                                                |
|---------------|-----------|--------|----------------------------------------------------------------------------------------------------------------------------|
| manifest_path | Yes       | String | Path for storing a manifest file, which can be a local path or an OBS path. If an OBS path is used, a session is required. |
| encoding      | No        | String | File encoding format, which defaults to UTF-8.                                                                             |

**Table 7-25** manifest\_info parameters

| Parameter | Type       | Description                                                |
|-----------|------------|------------------------------------------------------------|
| size      | Long       | Number of samples.                                         |
| samples   | JSON Array | Sample list. For details, see <a href="#">Table 7-26</a> . |

**Table 7-26 sample** parameters

| Parameter        | Type       | Description                                                                                                                                                                                                                           |
|------------------|------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| source           | String     | URI of the labeled object. Supported schemes are <b>OBS</b> , <b>HTTPS</b> , and <b>Content</b> . <b>Content</b> indicates text, for example, "source": "s3://path-to-jpg" and "source": " content://I love machine learning".        |
| annotations      | JSON Array | Sample labels. If this parameter is not specified, the object is not labeled.<br>The <b>annotations</b> value is an object list. For details, see <a href="#">Table 7-27</a> .                                                        |
| usage            | String     | What an object is used for, which can be training ( <b>TRAIN</b> ), evaluation ( <b>EVAL</b> ), test ( <b>TEST</b> ), or inference ( <b>INFERENCE</b> ). If this parameter is not specified, you can determine how to use the object. |
| inference_loc    | String     | Location of an inference result file. This parameter is available if a manifest file is generated in an inference service.                                                                                                            |
| id               | String     | Sample ID.                                                                                                                                                                                                                            |
| source_type      | String     | Source type, for example, <b>csv</b> .                                                                                                                                                                                                |
| source_property  | String     | Attribute of the source.                                                                                                                                                                                                              |
| hard             | Boolean    | Hard example or not. <b>true</b> for hard examples and <b>false</b> for not.                                                                                                                                                          |
| hard_coefficient | Double     | Difficulty coefficient, ranging from 0 to 1.                                                                                                                                                                                          |
| hard_reasons     | String     | Label-level hard example reasons. Use a hyphen (-) to separate reason IDs of a hard example.                                                                                                                                          |
| source_map       | String     | Source mapping.                                                                                                                                                                                                                       |

**Table 7-27 annotation** parameters

| Parameter      | Type   | Description                                                                                                   |
|----------------|--------|---------------------------------------------------------------------------------------------------------------|
| name           | String | Label name                                                                                                    |
| type           | String | Label type                                                                                                    |
| id             | String | Label ID                                                                                                      |
| annotation_loc | String | Location where a labeled file is stored. This parameter is mandatory only for object detection labeled files. |

| Parameter           | Type    | Description                                                                                                                    |
|---------------------|---------|--------------------------------------------------------------------------------------------------------------------------------|
| annotation_property | String  | Label properties                                                                                                               |
| confidence          | Double  | Confidence of machine labeling, which is a numeral ranging from 0 to 1                                                         |
| creation_time       | String  | Time when a label was created, which is the time when the label was written, not the time when the manifest file was generated |
| annotated_by        | String  | Annotator                                                                                                                      |
| annotation_format   | String  | Format of a labeled file, which defaults to <b>PASCAL VOC</b>                                                                  |
| hard                | Boolean | Hard example                                                                                                                   |
| hard_coefficient    | Double  | Difficulty level                                                                                                               |
| annotation_loc_map  | String  | Mapping of the path for storing a labeled file                                                                                 |

### 7.6.3 Creating and Saving a Manifest File

Create an object that contains the manifest information and save the object. For details about the manifest information, see [Table 7-25](#). The path can be either a local or OBS path. If an OBS path is used, a session is required.

```
manifest_info.save(path, session=None, save_mode="w")
```

#### Sample Code

Before saving a manifest file, create an object that contains the manifest information, including the samples and their labels, and then combine the samples into the manifest file. Call the **save** API to save the imported session in a specified path.

```
from modelarts.dataset.format.manifest.annotation import Annotation
from modelarts.dataset.format.manifest import Manifest
from modelarts.dataset.format.manifest.sample import Sample
from modelarts.session import Session

size = 0
sample_list = []
for i in range(19):
 size = size + 1
 source = "s3://obs-path/examples/image-classification/data/image_" + str(i) + ".jpg"
 usage = "TRAIN"
 inference_loc = "s3://obs-path/examples/image-classification/data/image_" + str(i) + ".txt"
 annotations_list = []

 for j in range(1):
 annotation_type = "modelarts/image_classification"
 if 0 == i % 2:
 annotation_name = "Bees"
 else:
 annotation_name = "Rabbits"
 annotation_creation_time = "2019-02-20 08:23:06"
```



```

annotation_format = "manifest"
annotation_property = {"color": "black"}
annotation_confidence = 0.8
annotated_by = "human"
annotations_list.append(
 Annotation(name=annotation_name, type=annotation_type,
 confidence=annotation_confidence,
 creation_time=annotation_creation_time,
 annotated_by=annotated_by, annotation_format=annotation_format,
 annotation_property=annotation_property))
sample_list.append(
 Sample(source=source, usage=usage, annotations=annotations_list, inference_loc=inference_loc))
manifest_info = Manifest(samples=sample_list, size=size)

path = "obs://your-obs-bucket/manifest/V001.manifest"
session = Session()
manifest_info.save(path, session=session, save_mode="a")

```

## Parameters

**Table 7-28** Request parameters

| Parameter | Mandatory | Type   | Description                                                                                                                                             |
|-----------|-----------|--------|---------------------------------------------------------------------------------------------------------------------------------------------------------|
| path      | Yes       | String | Path for storing a manifest file                                                                                                                        |
| session   | No        | Object | Session object. For details about the initialization method, see <a href="#">Session Authentication</a> . This parameter is mandatory when OBS is used. |
| save_mode | No        | String | Save mode. The default value is <b>w</b> , indicating rewriting. Value <b>a</b> indicates appending.                                                    |

### 7.6.4 Parsing a Pascal VOC File

Parse an XML file in either a local or OBS path. If an OBS path is used, a session is required.

```
PascalVoc.parse_xml(xml_file_path, session=None)
```

#### Sample Code

Specify an XML file path and call **parse\_xml** to parse the file.

```

from modelarts.dataset.format.voc.pascal_voc import PascalVoc
from modelarts.session import Session

path = "obs://your-obs-bucket/voc/test.xml"
session = Session()
pascal_voc = PascalVoc.parse_xml(path, session=session)
print(pascal_voc) # Print the parsing result.

```

## Parameters

**Table 7-29** Request parameters

| Parameter     | Mandatory | Type   | Description                                                                                                                                             |
|---------------|-----------|--------|---------------------------------------------------------------------------------------------------------------------------------------------------------|
| xml_file_path | Yes       | String | XML file path                                                                                                                                           |
| session       | No        | Object | Session object. For details about the initialization method, see <a href="#">Session Authentication</a> . This parameter is mandatory when OBS is used. |

**Table 7-30** pascal\_voc parameters

| Parameter   | Type       | Description                                                                                       |
|-------------|------------|---------------------------------------------------------------------------------------------------|
| folder      | String     | Folder name                                                                                       |
| file_name   | String     | File name                                                                                         |
| source      | Object     | Data source. For details, see <a href="#">Table 7-31</a> .                                        |
| width       | Long       | Image width                                                                                       |
| height      | Long       | Image height                                                                                      |
| depth       | Long       | Image depth                                                                                       |
| segmented   | String     | Segmentation                                                                                      |
| mask_source | String     | Path for storing the mask file generated after image segmentation. Only PNG images are supported. |
| voc_objects | JSON Array | Labeled objects. For details, see <a href="#">Table 7-32</a> .                                    |

**Table 7-31** source parameters

| Parameter  | Type   | Description                                            |
|------------|--------|--------------------------------------------------------|
| database   | String | Dataset name, for example, <b>The VOC2007 Database</b> |
| annotation | String | Label, for example, <b>PASCAL VOC2007</b>              |
| image      | String | Image information                                      |

**Table 7-32 voc\_object** parameters

| Parameter  | Type       | Description                                                                                                  |
|------------|------------|--------------------------------------------------------------------------------------------------------------|
| name       | String     | Folder name                                                                                                  |
| properties | JSON Array | Properties of a labeled object in key-value pairs. Both key and value are of the string type.                |
| pose       | String     | Shooting angle of labeled data                                                                               |
| truncated  | String     | Whether a labeled object is truncated ( <b>0</b> indicates the object is not truncated.)                     |
| occluded   | String     | Whether a labeled object is occluded ( <b>0</b> indicates the object is not occluded.)                       |
| difficult  | String     | Whether a labeled object is difficult to identify ( <b>0</b> indicates that the object is easy to identify.) |
| confidence | Double     | Confidence of machine labeling, which is a numeral ranging from 0 to 1                                       |
| position   | Object     | Location of a labeled object. For details, see <a href="#">Table 7-33</a> .                                  |
| parts      | Object     | Built-in voc_object list. For details, see <a href="#">Table 7-32</a> .                                      |
| mask_color | String     | Color of the mask image for image segmentation                                                               |

**Table 7-33 Position** parameters

| Parameter | Shape     | Labeling Information                                                                                                                     |
|-----------|-----------|------------------------------------------------------------------------------------------------------------------------------------------|
| point     | Point     | Coordinates of a point<br><x>100<x><br><y>100<y>                                                                                         |
| line      | Line      | Coordinates of points<br><x1>100<x1><br><y1>100<y1><br><x2>200<x2><br><y2>200<y2>                                                        |
| bndbox    | Rectangle | Coordinates of the lower left and upper right points<br><x_min>100<x_min><br><y_min>100<y_min><br><x_max>200<x_max><br><y_max>200<y_max> |

| Parameter | Shape   | Labeling Information                                                                                                                                                                                     |
|-----------|---------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| polygon   | Polygon | Coordinates of points<br><x1>100<x1><br><y1>100<y1><br><x2>200<x2><br><y2>100<y2><br><x3>250<x3><br><y3>150<y3><br><x4>200<x4><br><y4>200<y4><br><x5>100<x5><br><y5>200<y5><br><x6>50<x6><br><y6>150<y6> |
| circle    | Circle  | Center coordinates and radius<br><cx>100<cx><br><cy>100<cy><br><r>50<r>                                                                                                                                  |

## 7.6.5 Creating and Saving a Pascal VOC File

Create an object that contains the Pascal VOC information and save the object. For details about Pascal VOC, see [Table 7-30](#). The path can be either a local or OBS path. If an OBS path is used, a session is required.

```
pascal_voc.save_xml(xml_file_path, save_mode='w', session=None)
```

### Sample Code

Before saving a Pascal VOC XML file, create an object that contains the Pascal VOC information, including the VOC object. Call the **save\_xml** API to save the imported session in a specified path.

```
from modelarts.dataset.format.voc.pascal_voc import PascalVoc
from modelarts.dataset.format.voc.voc_object import VocObject
from modelarts.session import Session

path = "obs://your-obs-bucket/voc/test2.xml"
size_list = [640, 321, 3]
file_name = "000000089955.jpg"
voc_object_tags = ["trafficlight", "trafficlight"]
voc_object_properties = [{"@modelarts:color": "#FFFFFF0", "@modelarts:shortcut": "C",
 "pose": "0", "truncated": "0", "difficult": "0",
 "@modelarts:shape": "bndbox", "@modelarts:feature": [[347, 186], [382, 249]]},
 {"@modelarts:color": "#FFFE0", "@modelarts:shortcut": "D",
 "pose": "0", "truncated": "0", "difficult": "0",
 "@modelarts:shape": "bndbox", "@modelarts:feature": [[544, 50], [591, 149]]}]
voc_objects = []
for i in range(len(voc_object_tags)):
 object_tag = voc_object_tags[i]
```

```

object_properties = voc_object_properties[i]
voc_objects.append(VocObject(name=object_tag, properties=object_properties))

pascal_voc = PascalVoc(file_name=file_name, width=size_list[0], height=size_list[1], depth=size_list[2],
 voc_objects=voc_objects)
session = Session()
pascal_voc.save_xml(path, session=session)

```

## Parameters

**Table 7-34** Request parameters

| Parameter     | Mandatory | Type   | Description                                                                                                                                             |
|---------------|-----------|--------|---------------------------------------------------------------------------------------------------------------------------------------------------------|
| xml_file_path | Yes       | String | Path for storing a Pascal VOC XML file.                                                                                                                 |
| session       | No        | Object | Session object. For details about the initialization method, see <a href="#">Session Authentication</a> . This parameter is mandatory when OBS is used. |
| save_mode     | No        | String | Save mode. The default value is <b>w</b> , indicating rewriting. Value <b>a</b> indicates appending.                                                    |

## 7.7 Managing Labeling Jobs

### 7.7.1 Creating a Labeling Job

Create a labeling job based on a dataset.

```
dataset.create_label_task(self, task_name=None, task_type=None, **kwargs)
```

### Sample Code

Example 1: Create an object detection labeling job based on an image dataset.

```

from modelarts.session import Session
from modelarts.dataset import Dataset

session = Session()
dataset = Dataset(session, dataset_id="VukxA2FlaTUm7tkDtq0") # Initialize the dataset.
create_task_resp = dataset.create_label_task(task_name="obj_detection_task", task_type=1,
description="label task")

```

## Parameters

**Table 7-35** Request parameters

| Parameter   | Mandatory | Type    | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |
|-------------|-----------|---------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| task_name   | Yes       | String  | Name of a labeling job                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
| task_type   | Yes       | Integer | Type of a labeling job. Options: <ul style="list-style-type: none"> <li>• <b>0</b>: image classification</li> <li>• <b>1</b>: object detection</li> <li>• <b>3</b>: image segmentation</li> <li>• <b>100</b>: text classification</li> <li>• <b>101</b>: named entity recognition</li> <li>• <b>102</b>: text triplet</li> <li>• <b>200</b>: sound classification</li> <li>• <b>201</b>: speech content</li> <li>• <b>202</b>: speech paragraph labeling</li> <li>• <b>400</b>: table dataset</li> <li>• <b>600</b>: video labeling</li> <li>• <b>900</b>: custom format</li> </ul> |
| description | No        | String  | Description of a labeling job                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |

### 7.7.2 Obtaining the Labeling Job List of a Dataset

Obtain the labeling job list of a dataset.

```
dataset.get_label_tasks(is_workforce_task=False, **kwargs)
```

#### Sample Code

- Example 1: Obtain all labeling jobs of a dataset and sort the jobs by creation time in descending order.

```
from modelarts.session import Session
from modelarts.dataset import Dataset

session = Session()
dataset = Dataset(session, dataset_id="VukxA2FlaTU7tkDtq0")
list_label_task_resp = dataset.get_label_tasks(sort_key="create_time", sort_dir="desc")
print(list_label_task_resp)
```
- Example 2: Obtain all team labeling jobs of a dataset.

```
list_label_task_resp = dataset.get_label_tasks(is_workforce_task=True)
print(list_label_task_resp)
```

## Parameters

**Table 7-36** Request parameters

| Parameter         | Mandatory | Type    | Description                                                                                                                                                                                                                                                |
|-------------------|-----------|---------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| is_workforce_task | No        | Boolean | Filter criteria, specifying whether to obtain only team labeling jobs <ul style="list-style-type: none"> <li>• <b>True</b>: Only team labeling jobs are obtained.</li> <li>• <b>False</b>: Obtain all labeling jobs. This is the default value.</li> </ul> |
| sort_key          | No        | String  | Field for sorting. Options: <ul style="list-style-type: none"> <li>• <b>create_time</b>: Sort jobs by creation time.</li> <li>• <b>task_name</b>: Sort jobs by job name.</li> </ul>                                                                        |
| sort_dir          | No        | String  | Sorting method. Options: <ul style="list-style-type: none"> <li>• <b>asc</b>: Labeling jobs are sorted in ascending order.</li> <li>• <b>desc</b>: Labeling jobs are sorted in descending order. This is the default value.</li> </ul>                     |

### 7.7.3 Obtaining Details About a Labeling Job

Obtain details about a labeling job.

```
dataset.get_label_task_info(task_id=None)
```

#### Sample Code

Obtain details about a labeling job.

```
task_info = dataset.get_label_task_info(task_id="xs9ZKzLluKzccQfsyi2")
print(task_info)
```

## Parameters

**Table 7-37** Request parameters

| Parameter | Mandatory | Type   | Description          |
|-----------|-----------|--------|----------------------|
| task_id   | Yes       | String | ID of a labeling job |

# 8 Training Management

## 8.1 Training Jobs

### 8.1.1 Creating a Training Job

#### Sample Code

In ModelArts notebook, you do not need to enter authentication parameters for session authentication. For details about session authentication of other development environments, see [Session Authentication](#).

#### NOTE

ModelArts SDK cannot be used to create training jobs using algorithms subscribed to in AI Gallery.

- Example 1: **Create a training job using a common AI engine.**

If both **framework\_type** and **framework\_version** are specified in estimator, a training job will be created using a common AI engine.

```
from modelarts.session import Session
from modelarts.train_params import TrainingFiles
from modelarts.train_params import OutputData
from modelarts.train_params import InputData
from modelarts.estimatorV2 import Estimator
session = Session()
Parameters received in the training script (set based on the site requirements):

parameters = [{"name": "mod", "value": "gpu"},
 {"name": "epoc_num", "value": "2"}]
estimator = Estimator(session=session,
 training_files=TrainingFiles(code_dir= "obs://bucket_name/code_dir/",
 boot_file="boot_file.py"),
 outputs=[OutputData(obs_path="obs://bucket_name/output/", name="output_dir")],
 parameters=parameters,
 framework_type='PyTorch', # Common AI engine
 framework_version='PyTorch-1.4.0-python3.6', # Version of the AI engine
 train_instance_type="modelarts.p3.large.public",
 train_instance_count=1,
 log_url="obs://bucket_name/log/",
 env_variables={"USER_ENV_VAR": "customize environment variable"},
 working_dir="/home/ma-user/modelarts/user-job-dir",
```



```
 local_code_dir="/home/ma-user/modelarts/user-job-dir",
 job_description="This is an image net train job")
job_instance = estimator.fit(inputs=[InputData(obs_path="obs://bucket_name/input/",
name="data_url")],
 job_name="job_name_1")
```

- **Example 2: Create a training job using a custom image.**

If both **user\_image\_url** and **user\_command** are specified in estimator, a training job will be created using a custom image and started using a custom boot command.

```
from modelarts.session import Session
from modelarts.train_params import TrainingFiles
from modelarts.train_params import OutputData
from modelarts.train_params import InputData
from modelarts.estimatorV2 import Estimator
session = Session()
Parameters received in the training script (set based on the site requirements):

parameters = [{"name": "mod", "value": "gpu"},
 {"name": "epoc_num", "value": 2}]
estimator = Estimator(session=session,
 training_files=TrainingFiles(code_dir="obs://bucket_name/code_dir/",
boot_file="boot_file.py"),
 outputs=[OutputData(obs_path="obs://bucket_name/output/", name="output_dir")],
 parameters=parameters,
 user_image_url="sdk-test/pytorch1_4:1.0.1", # URL of the custom image
 user_command="/home/ma-user/anaconda3/envs/PyTorch-1.4/bin/python /home/ma-
user/modelarts/user-job-dir/train/test-pytorch.py", # Custom boot command
 train_instance_type="modelarts.p3.large.public",
 train_instance_count=1,
 log_url="obs://bucket_name/log/",
 env_variables={"USER_ENV_VAR": "customize environment variable"},
 working_dir="/home/ma-user/modelarts/user-job-dir",
 local_code_dir="/home/ma-user/modelarts/user-job-dir",
 job_description="This is an image net train job")
job_instance = estimator.fit(inputs=[InputData(obs_path="obs://bucket_name/input/",
name="data_url")],
 job_name="job_name_2")
```

- **Example 3: Creating a training job in a dedicated resource pool**

```
from modelarts.session import Session
from modelarts.train_params import TrainingFiles
from modelarts.train_params import OutputData
from modelarts.train_params import InputData
from modelarts.estimatorV2 import Estimator
session = Session()
Parameters received in the training script (set based on the site requirements):

parameters = [{"name": "mod", "value": "gpu"},
 {"name": "epoc_num", "value": 2}]
estimator = Estimator(session=session,
 training_files=TrainingFiles(code_dir="obs://bucket_name/code_dir/",
boot_file="boot_file.py"),
 outputs=[OutputData(obs_path="obs://bucket_name/output/", name="output_dir")],
 parameters=parameters,
 framework_type='PyTorch',
 framework_version='PyTorch-1.4.0-python3.6',
 pool_id="your pool id", # Dedicated resource pool ID
 train_instance_type="modelarts.pool.visual.xlarge", # VM flavor of the dedicated
pool
 train_instance_count=1,
 log_url="obs://bucket_name/log/",
 env_variables={"USER_ENV_VAR": "customize environment variable"},
 working_dir="/home/ma-user/modelarts/user-job-dir",
 local_code_dir="/home/ma-user/modelarts/user-job-dir",
 job_description="This is an image net train job")
job_instance = estimator.fit(inputs=[InputData(obs_path="obs://bucket_name/input/",
name="data_url")],
 job_name="job_name_3")
```

- **Example 4: Create a training job using a dataset.**

```

from modelarts.session import Session
from modelarts.train_params import TrainingFiles
from modelarts.train_params import OutputData
from modelarts.train_params import InputData
from modelarts.estimatorV2 import Estimator
session = Session()
Parameters received in the training script (set based on the site requirements):
parameters = [{"name": "model_name", "value": "s"},
 {"name": "batch-size", "value": 32},
 {"name": "epochs", "value": 100},
 {"name": "img-size", "value": "640,640"}]
estimator = Estimator(session=session,
 training_files=TrainingFiles(code_dir= "obs://bucket_name/code_dir/",
 boot_file="boot_file.py"),
 outputs=[OutputData(obs_path="obs://bucket_name/output/", name="output_dir")],
 parameters=parameters,
 framework_type='PyTorch', # Common AI engine
 framework_version='PyTorch-1.4.0-python3.6', # Version of the AI engine
 train_instance_type="modelarts.p3.large.public",
 train_instance_count=1,
 log_url="obs://bucket_name/log/",
 working_dir="/home/ma-user/modelarts/user-job-dir",
 local_code_dir="/home/ma-user/modelarts/user-job-dir",
 job_description='This is an image net train job')
job_instance = estimator.fit(dataset_id="your dataset id",
 dataset_version_id="your dataset version id",
 job_name="job_name_5")

```

## Parameters

**Table 8-1** Estimator request parameters

| Parameter            | Mandatory | Type                                        | Description                                                                                                                                              |
|----------------------|-----------|---------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------|
| session              | Yes       | Object                                      | Session object. For details about the initialization method, see <a href="#">Session Authentication</a> .                                                |
| training_files       | No        | <a href="#">TrainingFiles</a> Object        | Path to the training script in OBS. For details, see <a href="#">Table 8-2</a> .                                                                         |
| outputs              | No        | Array of <a href="#">OutputData</a> objects | Training output path. For details, see <a href="#">Table 8-3</a> .                                                                                       |
| parameters           | No        | JSON Array                                  | Running parameters of a training job. The format is as follows: [{"name": "your name", "value": "your value"}]. The value can be a string or an integer. |
| train_instance_type  | Yes       | String                                      | Resource flavor selected for a training job. For details, see <a href="#">Obtaining Resource Flavors</a> .                                               |
| train_instance_count | Yes       | Int                                         | Number of compute nodes in a training job                                                                                                                |

| Parameter         | Mandatory | Type       | Description                                                                                                                                                                                                                                                                                                                                                                  |
|-------------------|-----------|------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| framework_type    | No        | String     | Engine type selected for a training job. For details, see <a href="#">Obtaining Engine Types</a> .                                                                                                                                                                                                                                                                           |
| framework_version | No        | String     | Engine version selected for a training job. For details, see <a href="#">Obtaining Engine Types</a> .                                                                                                                                                                                                                                                                        |
| user_image_url    | No        | String     | SWR URL of the custom image used by a training job                                                                                                                                                                                                                                                                                                                           |
| user_command      | No        | String     | Command for starting a training job created using a custom image                                                                                                                                                                                                                                                                                                             |
| log_url           | No        | String     | OBS path for storing training job logs, for example, <b>obs://xx/yy/zz/</b>                                                                                                                                                                                                                                                                                                  |
| local_code_dir    | No        | String     | Local directory to the training container to which the algorithm code directory is downloaded. Note: <ul style="list-style-type: none"> <li>The directory must be under <b>/home</b>.</li> <li>In v1 compatibility mode, this parameter does not take effect.</li> <li>When <b>code_dir</b> is prefixed with <b>file://</b>, this parameter does not take effect.</li> </ul> |
| working_dir       | No        | String     | Work directory where an algorithm is executed. Note that this parameter does not take effect in v1 compatibility mode.                                                                                                                                                                                                                                                       |
| job_description   | No        | String     | Description of a training job                                                                                                                                                                                                                                                                                                                                                |
| volumes           | No        | JSON Array | Information of the disks attached for a training job in the following example format: <pre>[{   "nfs": {     "local_path": "/xx/yy/zz",     "read_only": False,     "nfs_server_path": "xxx.xxx.xxx.xxx:/"   } }]</pre>                                                                                                                                                      |
| env_variables     | No        | Dict       | Environment variables of a training job                                                                                                                                                                                                                                                                                                                                      |

| Parameter | Mandatory | Type   | Description                                                                                                                                                                                                                                                            |
|-----------|-----------|--------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| pool_id   | No        | String | ID of the resource pool for a training job. To obtain the ID, do as follows: Log in to the ModelArts management console, choose <b>Dedicated Resource Pools</b> in the navigation pane on the left, and view the resource pool ID in the dedicated resource pool list. |

**Table 8-2** Parameters for initializing **TrainingFiles**

| Parameter | Mandatory | Type   | Description                                                                                                                                                                                                    |
|-----------|-----------|--------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| code_dir  | Yes       | String | Code directory of a training job, which is an OBS path and must start with <b>obs:/</b> , for example, <b>obs://xx/yy/</b>                                                                                     |
| boot_file | Yes       | String | Boot file of a training job, which must be stored in the code directory. You can enter a relative path, for example, <b>boot_file.py</b> , or an absolute path, for example, <b>obs://xx/yy/boot_file.py</b> . |

**Table 8-3** Parameters for initializing **OutputData**

| Parameter | Mandatory | Type   | Description                                                               |
|-----------|-----------|--------|---------------------------------------------------------------------------|
| obs_path  | Yes       | String | OBS path to which data is exported                                        |
| name      | Yes       | String | Keyword parameter name of the output data, for example, <b>output_dir</b> |

**Table 8-4** fit request parameters

| Parameter | Mandatory | Type                             | Description                                                                                                                 |
|-----------|-----------|----------------------------------|-----------------------------------------------------------------------------------------------------------------------------|
| inputs    | No        | Array of <b>InputData</b> Object | Input data of a training job stored in OBS. Either <b>inputs</b> or <b>dataset_id/dataset_version_id</b> can be configured. |

| Parameter          | Mandatory | Type    | Description                                                                                                                                                                           |
|--------------------|-----------|---------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| wait               | No        | Boolean | Whether to wait for the completion of a training job. It defaults to <b>False</b> .                                                                                                   |
| job_name           | No        | String  | Name of a training job                                                                                                                                                                |
| show_log           | No        | Boolean | Whether to output training job logs after a job is submitted. It defaults to <b>False</b> .                                                                                           |
| dataset_id         | No        | String  | Dataset ID of a training job. For details, see <a href="#">Data Management</a> . This parameter must be used with <b>dataset_version_id</b> , but cannot be used with <b>inputs</b> . |
| dataset_version_id | No        | String  | Dataset version ID of a training job. For details, see <a href="#">Data Management</a> . This parameter must be used with <b>dataset_id</b> , but cannot be used with <b>inputs</b> . |

**Table 8-5** Parameters for initializing **InputData**

| Parameter | Mandatory | Type   | Description                                                                          |
|-----------|-----------|--------|--------------------------------------------------------------------------------------|
| obs_path  | Yes       | String | OBS path to the dataset required by a training job, for example, <b>obs://xx/yy/</b> |
| name      | Yes       | String | Keyword parameter name of the input data, for example, <b>data_url</b> .             |

**Table 8-6** Response for creating a training job

| Parameter   | Type   | Description                                                                                                                                                                                                                                                          |
|-------------|--------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| TrainingJob | Object | Training object, which contains attributes such as <b>job_id</b> . When you perform operations on a training job, for example, obtain information of, update, or delete a training job, you can use <b>job_instance.job_id</b> to obtain the ID of the training job. |

**Table 8-7** Response for the failure to call a training API

| Parameter      | Type   | Description                                                                                                                                                               |
|----------------|--------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| error_msg      | String | Error message when calling an API failed. This parameter is unavailable if an API is successfully called.                                                                 |
| error_code     | String | Error code when calling an API failed. For details, see "Error Codes" in <i>ModelArts API Reference</i> . This parameter is unavailable if an API is successfully called. |
| error_solution | String | Solution to an API calling failure. This parameter is unavailable if an API is successfully called.                                                                       |

## 8.1.2 Debugging a Training Job

### 8.1.2.1 Using the SDK to Debug a Multi-Node Distributed Training Job

Replace the OBS paths in the debugging code with your OBS paths.

PyTorch is used to write debugging code in this document. The process is the same for different AI frameworks. You only need to change the **framework\_type** value in [Step 7](#) and [Step 11](#). For example, set **framework\_type** to **Ascend-Powered-Engine** for MindSpore.

**Step 1** Initialize session. This step is the same as that of [debugging a single-node training job](#).

**Step 2** Prepare training data. This step is the same as that of [debugging a single-node training job](#). The only difference is that **obs\_path** must be set here.

**Step 3** Prepare the training script.

```
from modelarts.train_params import TrainingFiles
code_dir = os.path.join(base_local_path, "train/")
```

```
The training script has been stored in OBS. The training script can be chosen from any source as long as it can be stored in a notebook instance.
```

```
session.obs.download_file(os.path.join(base_bucket_path, "train/test-pytorch.py"), code_dir)
training_file = TrainingFiles(code_dir=code_dir, boot_file="test-pytorch.py", obs_path=base_bucket_path + 'train/')
```

Parameters:

- **code\_dir**: Code directory where a training script is stored. The directory must be a notebook directory for local debugging. This parameter is mandatory.
- **boot\_file**: Training boot file, which is stored in the **code\_dir** directory. This parameter is mandatory.
- **obs\_path**: OBS directory. This parameter is mandatory for multi-node distributed debugging. The SDK zips the notebook directory **code\_dir** and uploads the ZIP file to **obs\_path**.

- Step 4** Prepare the training output. This step is the same as [Step 4](#) for debugging a single-node training job.
- Step 5** Check the AI frameworks available for training. This step is the same as [Step 5](#) for debugging a single-node training job.
- Step 6** Save the current notebook instance as a new image. This step is the same as [Step 9](#) for debugging a single-node training job.
- Step 7** Initialize the Estimator.

```
from modelarts.estimatorV2 import Estimator
parameters = []
parameters.append({"name": "data_url", "value": data_local})
parameters.append({"name": "output_dir", "value": os.path.join(base_local_path, "output/")})
parameters.append({"name": "epoc_num", "value": 2})
For Boolean, use parser.add_argument('--dist', action='store_true') in the boot script for parsing. If the
parameter is set to True, the parameter is transferred in the format of the following lines of code.
parameters.append({"name": "dist"})
estimator = Estimator(session=session,
 training_files=training_file,
 outputs=[output],
 parameters=parameters,
 framework_type='PyTorch',
 train_instance_type='local',
 train_instance_count=2,
 script_interpreter="/home/ma-user/anaconda3/envs/PyTorch-1.4/bin/python",
 log_url=base_bucket_path + 'log/',
 job_description='This is a image net train job')
```

Parameters:

- **session**: Initialized data in [Step 1](#). This parameter is mandatory.
- **training\_files**: Initialized training files in [Step 3](#). This parameter is mandatory.
- **outputs**: A list of training outputs. Each element in the list is a training output initialized in [Step 4](#). This parameter is optional.
- **parameters**: A list of parameters. Each element in the list is a dictionary that contains the **name** and **value** fields, which are transferred to the training boot file in the form of **-name=value**. This parameter is optional. **value** can be a string, an integer, or a Boolean. For Boolean, use **action='store\_true'** in the training script for parsing.
- **framework\_type**: Type of the AI framework used for a training job. For details, see the output item in [Step 5](#). This parameter is mandatory.
- **train\_instance\_type**: Type of training instance. If this parameter is set to **local**, the training job is performed in a notebook instance. This parameter is mandatory.
- **train\_instance\_count**: Number of workers in a training job. Set this parameter to **2** for distributed debugging. When the training job starts, the SDK creates another notebook instance to form a 2-node distributed debugging environment with the current instance. This parameter is mandatory.
- **script\_interpreter**: Python environment used for a training job. If this parameter is not set, the current kernel is used by default. This parameter is optional.
- **log\_url**: OBS address. The SDK automatically uploads training logs to this address during local training. This parameter must be set only when training jobs run on Ascend.

- **job\_description**: describes a training job. This parameter is optional.

**Step 8** Start training.

```
estimator.fit(inputs=[input_data], job_name="cifar10-dis")
```

Parameters:

- **inputs**: A list of training inputs. Each element in the list is an input imported in [Step 2](#). This parameter is optional.
- **job\_name**: Name of a training job. This parameter is optional.

After a local distributed training job starts, the SDK automatically performs the following operations:

1. Zips the training script and uploads the ZIP file to **obs\_path** specified in [Step 3](#).
2. Zips the data and uploads the ZIP file to the specified **obs\_path** if the training data is stored in .
3. Creates another instance to form a two-worker environment for distributed training.
4. Initializes the training job and downloads data to **local\_path**.
5. Executes the training job and saves the training outputs in **local\_path** specified in [Step 4](#).
6. Uploads the training output to **obs\_path** specified in [Step 4](#) and the logs to **log\_url** specified in [Step 7](#).

**Step 9** Perform debugging. This step is the same as [Step 8](#) for debugging a single-node training job.

**Step 10** Obtain the type of compute nodes available for training. This step is the same as [Step 9](#) for debugging a single-node training job.

**Step 11** Submit the remote training job.

```
from modelarts.estimatorV2 import Estimator
parameters = []
parameters.append({"name": "data_url", "value": data_local})
parameters.append({"name": "output_dir", "value": os.path.join(base_local_path, "output/")})
parameters.append({"name": "epoch_num", "value": 2})
For Boolean, use parser.add_argument('--dist', action='store_true') in the boot script for parsing. If the
parameter is set to True, the parameter is transferred in the format of the following lines of code.
parameters.append({"name": "dist"})
estimator = Estimator(session=session,
 training_files=training_file,
 outputs=[output],
 parameters=parameters,
 framework_type='PyTorch',
 train_instance_type='modelarts.p3.large.public.distributed',
 train_instance_count=2,
 script_interpreter="/home/ma-user/anaconda3/envs/PyTorch-1.4/bin/python",
 log_url=base_bucket_path + 'log/',
 job_description='This is a image net train job')
estimator.fit(inputs=[input_data], job_name="cifar10-dis-1")
```

The difference between Estimator initialization and local training lies in the **train\_instance\_type** parameter. Configure this parameter based on the obtained result in [Step 10](#). The value of **train\_instance\_count** depends on the value of **max\_num** in [Step 10](#).

After the training job is submitted, the SDK automatically performs the following operations:



1. Zips the training script and uploads the ZIP file to **obs\_path** specified in [Step 3](#).
2. Zips the data and uploads the ZIP file to the specified **obs\_path** if the training data is stored in .
3. Submits the training job to the ModelArts training service. The image of the current instance is used to execute the training job.
4. Uploads the training output to **obs\_path** specified in [Step 4](#) and the logs to the location specified by **log\_url**.

In this step, note the following:

If you want to create a directory or file in your training script, create it in the following directories:

(1) **/home/ma-user/work**

(2) **/cache**

(3) **local\_path** specified in inputs or outputs. For example, if **local\_path** is set to **/home/ma-user/work/xx/yy/** during InputData initialization in [Step 2](#), you can create directories or files in this directory.

----End

### 8.1.2.2 Using the SDK to Debug a Single-Node Training Job

Replace the OBS paths in the debugging code with your OBS paths.

PyTorch is used to write debugging code in this document. The process is the same for different AI frameworks. You only need to change the **framework\_type** value in [Step 6](#) and [Step 10](#). For example, set **framework\_type** to **Ascend-Powered-Engine** for MindSpore.

#### Step 1 Initialize session.

The following is the sample code.

```
from modelarts.session import Session
session = Session()
```

#### Step 2 Prepare training data. Three data formats are supported. You can select one of them as required.

```
import os
from modelarts.train_params import InputData
base_bucket_path = "obs://modelarts-xxx-a0de02a6/dis-train/cifar10/"
base_local_path = "/home/ma-user/work/cifar10/"

Format 1: The data is stored in a compressed file in OBS.
obs_path = os.path.join(base_bucket_path, "dataset-zip/dataset.zip")
data_local = os.path.join(base_local_path, "dataset/")
input_data = InputData(obs_path=obs_path, local_path=data_local, is_local_source=False)

Format 2: The data is stored in a directory in OBS.
#obs_path = os.path.join(base_bucket_path, "dataset/")
#data_local = os.path.join(base_local_path, "dataset/")
#input_data = InputData(obs_path=obs_path, local_path=data_local, is_local_source=False)

Format 3: The data is stored in a directory in an SFS system mounted to a notebook instance.
#obs_path = os.path.join(base_bucket_path, "dataset-local/")
#data_local = os.path.join(base_local_path, "dataset/")
#input_data = InputData(obs_path=obs_path, local_path=data_local, is_local_source=True)
```

Parameters:

- **is\_local\_source**: Location where the training data is stored. The default value is **False** and this parameter is optional.
  - **False**: The training data is stored in the path specified by **obs\_path**.
  - **True**: The training data is stored in a notebook instance, which is specified by **local\_path**.
- **obs\_path**: OBS path. It depends on the value of **is\_local\_source**.
  - If **is\_local\_source** is set to **False**, this parameter is mandatory, indicating the location where the training data is stored, which can be a folder or a compressed file.
  - If **is\_local\_source** is set to **True**, this parameter is optional. If you set this parameter, the training data in the notebook instance is compressed and uploaded to the location. The data cannot be uploaded repeatedly. After data is uploaded for the first time, change **is\_local\_source** to **False** and set **obs\_path** to the location where the compressed file was uploaded. If you do not set this parameter, the file will not be compressed and uploaded.
- **local\_path**: Notebook path. Your training script reads data from this path for training. This parameter is mandatory. It depends on the value of **is\_local\_source**.
  - If **is\_local\_source** is set to **True**, this parameter indicates the location where the training data is stored, which can be a folder.
  - If **is\_local\_source** is set to **False**, the SDK downloads data to this location during training. If the training data is compressed files, they will be decompressed after being downloaded.

### Step 3 Prepare the training script.

```
from modelarts.train_params import TrainingFiles
code_dir = os.path.join(base_local_path, "train/")

The training script has been stored in OBS. The training script can be chosen from any source as long as it
can be stored in a notebook instance.

session.obs.download_file(os.path.join(base_bucket_path, "train/test-pytorch.py"), code_dir)
training_file = TrainingFiles(code_dir=code_dir, boot_file="test-pytorch.py", obs_path=base_bucket_path +
'train/')
```

Parameters:

- **code\_dir**: Code directory where a training script is stored. The directory must be a notebook directory for debugging a training job. This parameter is mandatory.
- **boot\_file**: Path of the training boot file. Enter the relative path of **code\_dir**. For example, if the absolute path of **boot\_file** is **/home/ma-user/work/cifar10/train/test-pytorch.py**, set this parameter to **test-pytorch.py**. This parameter is mandatory.
- **obs\_path**: OBS path. This parameter must be set only for remote training. The training script is compressed and uploaded to this path.

### Step 4 Prepare the training output. If you do not need to upload the training output to OBS, skip this step.

```
from modelarts.train_params import OutputData
output = OutputData(local_path=os.path.join(base_local_path, "output/"),
obs_path=os.path.join(base_bucket_path, 'output/'))
```

- **local\_path**: Notebook path, in which the trained model or other training script data is stored.
- **obs\_path**: OBS path. The SDK automatically uploads the model file in **local\_path** to this OBS path. This parameter is mandatory.

#### Step 5 Check the AI frameworks that can be used for training.

```
from modelarts.estimatorV2 import Estimator
Estimator.get_framework_list(session)
```

**session** is the initialized data in [Step 1](#). Skip this step if the AI framework has been specified.

#### Step 6 Initialize the Estimator.

```
from modelarts.estimatorV2 import Estimator
parameters = []
parameters.append({"name": "data_url", "value": data_local})
parameters.append({"name": "output_dir", "value": os.path.join(base_local_path, "output/")})
parameters.append({"name": "epoch_num", "value": 2})
estimator = Estimator(session=session,
 training_files=training_file,
 outputs=[output],
 parameters=parameters,
 framework_type='PyTorch',
 train_instance_type='local',
 train_instance_count=1,
 script_interpreter="/home/ma-user/anaconda3/envs/PyTorch-1.4/bin/python",
 log_url=base_bucket_path + 'log/',
 job_description='This is a image net train job')
```

Parameters:

- **session**: Initialized data in [Step 1](#). This parameter is mandatory.
- **training\_files**: Initialized training files in [Step 3](#). This parameter is mandatory.
- **outputs**: A list of training outputs. Each element in the list is an initialized training output in [Step 4](#). This parameter is optional.
- **parameters**: A list of parameters. This parameter is optional. Each element in the list is a dictionary that contains the **name** and **value** fields, which are transferred to the training boot file in the form of **--name=value**. **value** can be a string, an integer, or a Boolean. For Boolean, use **action='store\_true'** in the training script for parsing.
- **framework\_type**: Type of the AI framework used for a training job. For details, see the output item in [Step 5](#). This parameter is mandatory.
- **train\_instance\_type**: Training instance type. If this parameter is set to **local**, the training job is performed in a notebook instance. This parameter is mandatory.
- **train\_instance\_count**: Number of workers in a training job. Set this parameter to **1** for single-node training. The training job runs only in the current notebook instance. This parameter is mandatory.
- **script\_interpreter**: Python environment used for a training job. If this parameter is not set, the current kernel is used by default. This parameter is optional.
- **log\_url**: OBS address. The SDK automatically uploads training logs to this address during training. This parameter must be set only when training jobs run on Ascend.
- **job\_description**: describes a training job. This parameter is optional.

**Step 7** Start training.

```
estimator.fit(inputs=[input_data], job_name="cifar10-dis")
```

Parameters:

- **inputs**: A list of training inputs. Each element in the list is an input imported in [Step 2](#). This parameter is optional.
- **job\_name**: Name of a training job. This parameter is optional.

After a local single-node training job starts, the SDK automatically performs the following operations:

1. Initializes the training job. If the training data imported in [Step 2](#) is stored in OBS, the data is downloaded to **local\_path**.
2. Executes the training job and saves the training outputs in **local\_path** specified in [Step 4](#).
3. Uploads the training output to **obs\_path** specified in [Step 4](#) and the logs to **log\_url** specified in [6](#).

In addition, time suffixes are added to the job names.

```
from datetime import datetime, timedelta
import time
base_name = "cifar10-dis"
job_name = base_name + '-' + (datetime.now() + timedelta(hours=8)).strftime('%Y%m%d-%H%M%S')
estimator.fit(inputs=[input_data], job_name=job_name)
```

**Step 8** Perform debugging.

In the previous step, the logs of the training script are printed to the console in real time. You can easily detect incorrect code or parameters in the logs. Perform debugging in Notebook until you obtain a desired result, then you can go to the next step.

**Step 9** Obtain the type and maximum number of compute nodes available for training.

```
from modelarts.estimatorV2 import Estimator
Estimator.get_spec_list(session=session)
```

**session** is the initialized data in [Step 1](#). A dictionary is returned. **flavors** is a list that describes all flavors available for training. **flavor\_id** of each element indicates the compute flavors that can be directly used for remote training jobs, and **max\_num** indicates the maximum number of compute nodes of the flavors. Skip this step if the compute flavor has been specified.

**Step 10** Submit the remote training job.

```
from modelarts.estimatorV2 import Estimator
parameters = []
parameters.append({"name": "data_url", "value": data_local})
parameters.append({"name": "output_dir", "value": os.path.join(base_local_path, "output/")})
parameters.append({"name": "epoch_num", "value": 2})
estimator = Estimator(session=session,
 training_files=training_file,
 outputs=[output],
 parameters=parameters,
 framework_type='PyTorch',
 train_instance_type='modelarts.vm.cpu.8u',
 train_instance_count=1,
 script_interpreter="/home/ma-user/anaconda3/envs/PyTorch-1.4/bin/python",
 log_url=base_bucket_path + 'log/',
 job_description='This is a image net train job')
estimator.fit(inputs=[input_data], job_name="cifar10-dis")
```

After the local debugging is complete, you only need to change **train\_instance\_type** to the value of **flavor\_id** obtained in [Step 9](#) during Estimator initialization. After the **fit** function is executed, you can submit the remote training job.

After the training job is submitted, the SDK automatically performs the following operations:

1. Zips the training script and uploads the ZIP file to **obs\_path** specified in [Step 3](#).
2. Zips the data and uploads the ZIP file to the specified **obs\_path** if the training data is stored in Notebook.
3. Submits the training job created using a custom image to ModelArts. The image is that of the current instance. This ensures that the environment of the remote training job is the same as that of the training job in the notebook instance.
4. Uploads the training output to **obs\_path** specified in [Step 4](#) and the logs to the location specified by **log\_url** in this step.

In this step, note the following:

If you want to create a directory or file in your training script, create it in the following directories:

- **/home/ma-user/work**
- **/cache**
- **local\_path** specified in inputs or outputs. For example, if **local\_path** is set to **/home/ma-user/work/xx/yy/** during InputData initialization in [Step 2](#), you can create directories or files in this directory.

----End

## 8.1.3 Obtaining Training Jobs

### Sample Code

In ModelArts notebook, you do not need to enter authentication parameters for session authentication. For details about session authentication of other development environments, see [Session Authentication](#).

```
from modelarts.session import Session
from modelarts.estimatorV2 import Estimator
session = Session()
job_list = Estimator.get_job_list(session=session, offset=10, limit=5, sort_by="create_time", order="asc",
 filters=[{"key": "name", "operator": "like", "value": ["trainjob"]})
print(job_list)
```

## Parameters

**Table 8-8** get\_job\_list request parameters

| Parameter | Mandatory | Type             | Description                                                                                                                                                                                                                                                                                                                                                                                                                     |
|-----------|-----------|------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| session   | Yes       | Object           | Session object. For details about the initialization method, see <a href="#">Session Authentication</a> .                                                                                                                                                                                                                                                                                                                       |
| offset    | No        | Integer          | Offset for obtaining training jobs. The minimum value is <b>0</b> . For example, if this parameter is set to <b>1</b> , the query starts from the second one.                                                                                                                                                                                                                                                                   |
| limit     | No        | Integer          | Maximum number of training jobs to be obtained. The value ranges from <b>1</b> to <b>50</b> .                                                                                                                                                                                                                                                                                                                                   |
| sort_by   | No        | String           | Metric for sorting obtained training jobs. By default, training jobs are sorted by creation time ( <b>create_time</b> ).                                                                                                                                                                                                                                                                                                        |
| order     | No        | String           | Order of obtained training jobs. The default value is <b>desc</b> , indicating the descending order. You can also set this parameter to <b>asc</b> , indicating the ascending order.<br>Default value: <b>desc</b><br>Options: <ul style="list-style-type: none"> <li>• <b>asc</b>: The query results are displayed in ascending order.</li> <li>• <b>desc</b>: The query results are displayed in descending order.</li> </ul> |
| group_by  | No        | String           | Condition for grouping the obtained training jobs.                                                                                                                                                                                                                                                                                                                                                                              |
| filters   | No        | Array of objects | Filter criteria for obtaining training jobs.                                                                                                                                                                                                                                                                                                                                                                                    |

**Table 8-9** filters

| Parameter | Mandatory | Type   | Description                    |
|-----------|-----------|--------|--------------------------------|
| key       | No        | String | Key of the grouping condition. |

| Parameter | Mandatory | Type             | Description                                                                                                                                                                                                                                                                |
|-----------|-----------|------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| operator  | No        | String           | The key-value relationship of a grouping condition.<br>Default value: <b>in</b><br>Options: <ul style="list-style-type: none"> <li>• <b>like</b>: similar</li> <li>• <b>in</b>: included</li> <li>• <b>not</b>: not included</li> <li>• <b>between</b>: a range</li> </ul> |
| value     | No        | Array of strings | Value of the grouping condition key.                                                                                                                                                                                                                                       |

**Table 8-10** get\_job\_list response parameters

| Parameter    | Type                                | Description                                                                                                                                                                          |
|--------------|-------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| total        | Integer                             | Total number of training jobs of the current user.                                                                                                                                   |
| count        | Integer                             | Total number of training jobs that meet the search criteria of the current user.                                                                                                     |
| limit        | Integer                             | Maximum number of training jobs to be obtained. The value ranges from <b>1</b> to <b>50</b> .                                                                                        |
| offset       | Integer                             | Offset for obtaining training jobs. The minimum value is <b>0</b> . For example, if this parameter is set to <b>1</b> , the query starts from the second one.                        |
| sort_by      | String                              | Metric for sorting obtained training jobs. By default, training jobs are sorted by creation time ( <b>create_time</b> ).                                                             |
| order        | String                              | Order of obtained training jobs. The default value is <b>desc</b> , indicating the descending order. You can also set this parameter to <b>asc</b> , indicating the ascending order. |
| group_by     | String                              | Condition for grouping the obtained training jobs.                                                                                                                                   |
| workspace_id | String                              | Workspace where a training job is deployed. The default value is <b>0</b> .                                                                                                          |
| ai_project   | String                              | AI project to which a training job belongs. The default value is <b>default-ai-project</b> .                                                                                         |
| items        | Array of <b>JobResponse</b> objects | Details of the training jobs that meet the search criteria of the current user.                                                                                                      |

**Table 8-11** JobResponse

| Parameter | Type                                 | Description                                                                                                                                                                                                                                                                                                                                                                                                      |
|-----------|--------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| kind      | String                               | Training job type, which defaults to <b>job</b> .<br>Options: <ul style="list-style-type: none"> <li>• <b>job</b>: training job</li> <li>• <b>hetero_job</b>: heterogeneous job</li> <li>• <b>autosearch_job</b>: auto search job</li> <li>• <b>mrs_job</b>: MRS job</li> <li>• <b>edge_job</b>: edge job</li> </ul>                                                                                             |
| metadata  | <b>JobMetadata</b> object            | Metadata of a training job.                                                                                                                                                                                                                                                                                                                                                                                      |
| status    | <b>Status</b> object                 | Status of a training job. When creating a training job, you do not need to set this parameter.                                                                                                                                                                                                                                                                                                                   |
| algorithm | <b>JobAlgorithmResponse</b> object   | Algorithm used by a training job. The following formats are supported: <ul style="list-style-type: none"> <li>• <b>id</b>: Only the algorithm ID is used.</li> <li>• <b>subscription_id</b> and <b>item_version_id</b>: The subscription ID and version ID of the algorithm are used.</li> <li>• <b>code_dir</b> and <b>boot_file</b>: The code directory and boot file of the training job are used.</li> </ul> |
| tasks     | Array of <b>TaskResponse</b> objects | Tasks of a heterogeneous training job.                                                                                                                                                                                                                                                                                                                                                                           |
| spec      | <b>spec</b> object                   | Specifications of a training job.                                                                                                                                                                                                                                                                                                                                                                                |

**Table 8-12** JobMetadata

| Parameter    | Type   | Description                                                                                                                             |
|--------------|--------|-----------------------------------------------------------------------------------------------------------------------------------------|
| id           | String | Training job ID, which is generated and returned by ModelArts after a training job is created.                                          |
| name         | String | Name of a training job. The value must contain 1 to 64 characters consisting of only digits, letters, underscores (_), and hyphens (-). |
| workspace_id | String | Workspace where a training job is deployed. Default value: <b>0</b>                                                                     |



| Parameter   | Type               | Description                                                                                                                                                                                |
|-------------|--------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| description | String             | Description of a training job, which defaults to <b>NULL</b> . The value must contain 0 to 256 characters.                                                                                 |
| create_time | Long               | Time when a training job was created, in milliseconds. The value is generated and returned by ModelArts after a training job is created.                                                   |
| username    | String             | Username for creating a training job. The username is generated and returned by ModelArts after a training job is created.                                                                 |
| annotations | Map<String,String> | Declaration template of a training job. For heterogeneous jobs, the default value of <b>job_template</b> is <b>Template RL</b> . For other jobs, the default value is <b>Template DL</b> . |

**Table 8-13** Status

| Parameter          | Type                  | Description                                                                                                                                                                                           |
|--------------------|-----------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| phase              | String                | Level-1 status of a training job. The value will remain unchanged. Options: <b>Creating, Pending, Running, Failed, Completed, Terminating, Terminated, and Abnormal</b>                               |
| secondary_phase    | String                | Level-2 status of a training job. The value can be changed. Options: <b>Creating, Queuing, Running, Failed, Completed, Terminating, Terminated, CreateFailed, TerminatedFailed, Unknown, and Lost</b> |
| duration           | Long                  | Running duration of a training job, in milliseconds                                                                                                                                                   |
| node_count_metrics | Array<Array<Integer>> | Node count changes during the runtime of a training job                                                                                                                                               |
| tasks              | Array of strings      | Tasks of a training job                                                                                                                                                                               |
| start_time         | String                | Start time of a training job. The value is in timestamp format.                                                                                                                                       |
| task_statuses      | Array of objects      | Status of a training job task                                                                                                                                                                         |

**Table 8-14** task\_statuses

| Parameter | Type    | Description                          |
|-----------|---------|--------------------------------------|
| task      | String  | Task of a training job               |
| exit_code | Integer | Exit code of a training job task     |
| message   | String  | Error message of a training job task |

**Table 8-15** JobAlgorithmResponse

| Parameter                 | Type   | Description                                                                                                                                                                                                                                                                                                                                                        |
|---------------------------|--------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| id                        | String | Algorithm ID<br>Options: <ul style="list-style-type: none"> <li>• <b>id</b>: Only the algorithm ID is used.</li> <li>• <b>subscription_id</b> and <b>item_version_id</b>: The subscription ID and version ID of the algorithm are used.</li> <li>• <b>code_dir</b> and <b>boot_file</b>: The code directory and boot file of the training job are used.</li> </ul> |
| name                      | String | Algorithm name                                                                                                                                                                                                                                                                                                                                                     |
| subscription_id           | String | Subscription ID of the subscribed algorithm, which must be used with <b>item_version_id</b>                                                                                                                                                                                                                                                                        |
| item_version_id           | String | Version ID of the subscribed algorithm, which must be used with <b>subscription_id</b>                                                                                                                                                                                                                                                                             |
| code_dir                  | String | Code directory of a training job, for example, <b>/usr/app/</b> . This parameter must be used with <b>boot_file</b> . Leave this parameter blank if <b>id</b> , or <b>subscription_id</b> and <b>item_version_id</b> are specified.                                                                                                                                |
| boot_file                 | String | Boot file of a training job, which must be stored in the code directory, for example, <b>/usr/app/boot.py</b> . This parameter must be used with <b>code_dir</b> . Leave this parameter blank if <b>id</b> , or <b>subscription_id</b> and <b>item_version_id</b> are specified.                                                                                   |
| autosearch_config_path    | String | YAML configuration path of an auto search job. An OBS URL is required.                                                                                                                                                                                                                                                                                             |
| autosearch_framework_path | String | Framework code directory of an auto search job. An OBS URL is required.                                                                                                                                                                                                                                                                                            |

| Parameter    | Type                              | Description                                                                                                                                                                                                            |
|--------------|-----------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| command      | String                            | Boot command for starting the container of the custom image used for creating a training job. The value of this parameter can be the same as the <b>code_dir</b> value.                                                |
| parameters   | Array of <b>Parameter</b> objects | Running parameters of a training job.                                                                                                                                                                                  |
| policies     | <b>policies</b> object            | Policies supported by a training job.                                                                                                                                                                                  |
| inputs       | Array of <b>Input</b> objects     | Input of a training job.                                                                                                                                                                                               |
| outputs      | Array of <b>Output</b> objects    | Output of a training job.                                                                                                                                                                                              |
| engine       | <b>engine</b> object              | Engine of a training job. Leave this parameter blank if the job is created using <b>id</b> of the algorithm in algorithm management, or <b>subscription_id</b> and <b>item_version_id</b> of the subscribed algorithm. |
| environments | Map<String,String>                | Environment variables of a training job in the format of "key":"value". Leave this parameter blank.                                                                                                                    |

**Table 8-16** Parameter

| Parameter        | Type                           | Description                      |
|------------------|--------------------------------|----------------------------------|
| name             | String                         | Parameter name                   |
| value            | String                         | Parameter value                  |
| description      | String                         | Parameter description            |
| constraint       | <b>constraint</b> object       | Parameter constraint             |
| i18n_description | <b>i18n_description</b> object | Internationalization description |

**Table 8-17** constraint

| Parameter   | Type             | Description                        |
|-------------|------------------|------------------------------------|
| type        | String           | Parameter type                     |
| editable    | Boolean          | Whether the parameter is editable  |
| required    | Boolean          | Whether the parameter is mandatory |
| sensitive   | Boolean          | Whether the parameter is sensitive |
| valid_type  | String           | Valid type                         |
| valid_range | Array of strings | Valid range                        |

**Table 8-18** i18n\_description

| Parameter   | Type   | Description                   |
|-------------|--------|-------------------------------|
| language    | String | Internationalization language |
| description | String | Description                   |

**Table 8-19** policies

| Parameter   | Type                               | Description                         |
|-------------|------------------------------------|-------------------------------------|
| auto_search | <a href="#">auto_search</a> object | Hyperparameter search configuration |

**Table 8-20** auto\_search

| Parameter          | Type             | Description                                       |
|--------------------|------------------|---------------------------------------------------|
| skip_search_params | String           | Hyperparameter parameters that need to be skipped |
| reward_attrs       | Array of objects | Search metrics                                    |

| Parameter     | Type             | Description                     |
|---------------|------------------|---------------------------------|
| search_params | Array of objects | Search parameters               |
| algo_configs  | Array of objects | Search algorithm configurations |

**Table 8-21** reward\_attrs

| Parameter | Type   | Description                                                                                                                                                                 |
|-----------|--------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| name      | String | Metric name                                                                                                                                                                 |
| mode      | String | Search mode <ul style="list-style-type: none"> <li>• <b>max</b>: A larger metric value is preferred.</li> <li>• <b>min</b>: A smaller metric value is preferred.</li> </ul> |
| regex     | String | Regular expression of a metric                                                                                                                                              |

**Table 8-22** search\_params

| Parameter           | Type             | Description                                                                                                                                                                       |
|---------------------|------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| name                | String           | Hyperparameter name                                                                                                                                                               |
| param_type          | String           | Parameter type <ul style="list-style-type: none"> <li>• <b>continuous</b>: Parameter values are continuous.</li> <li>• <b>discrete</b>: Parameter values are discrete.</li> </ul> |
| lower_bound         | String           | Lower bound of the hyperparameter                                                                                                                                                 |
| upper_bound         | String           | Upper bound of the hyperparameter                                                                                                                                                 |
| discrete_points_num | String           | Number of discrete points of a hyperparameter with continuous values                                                                                                              |
| discrete_values     | Array of strings | Discrete hyperparameter values                                                                                                                                                    |

**Table 8-23** algo\_configs

| Parameter | Type                                                  | Description                  |
|-----------|-------------------------------------------------------|------------------------------|
| name      | String                                                | Name of the search algorithm |
| params    | Array of <b>AutoSearchAlgoConfigParameter</b> objects | Search algorithm parameters  |

**Table 8-24** AutoSearchAlgoConfigParameter

| Parameter | Type   | Description     |
|-----------|--------|-----------------|
| key       | String | Parameter key   |
| value     | String | Parameter value |
| type      | String | Parameter type  |

**Table 8-25** Input

| Parameter         | Type                        | Description                                                                |
|-------------------|-----------------------------|----------------------------------------------------------------------------|
| name              | String                      | Name of the data input channel                                             |
| description       | String                      | Description of the data input channel                                      |
| local_dir         | String                      | Local directory of the container to which the data input channel is mapped |
| remote            | <b>InputDataInfo</b> object | Information of the data input                                              |
| remote_constraint | Array of objects            | Data input constraint                                                      |

**Table 8-26** InputDataInfo

| Parameter | Type                  | Description                                   |
|-----------|-----------------------|-----------------------------------------------|
| dataset   | <b>dataset</b> object | Dataset as the data input                     |
| obs       | <b>obs</b> object     | OBS in which data input and output are stored |

**Table 8-27** dataset

| Parameter  | Type   | Description                                                                                                                                                           |
|------------|--------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| id         | String | Dataset ID of a training job                                                                                                                                          |
| version_id | String | Dataset version ID of a training job                                                                                                                                  |
| obs_url    | String | OBS URL of the dataset for a training job, which is automatically parsed by ModelArts based on the dataset ID and dataset version IDs, for example, <b>/usr/data/</b> |

**Table 8-28** obs

| Parameter | Type   | Description                                                               |
|-----------|--------|---------------------------------------------------------------------------|
| obs_url   | String | OBS URL of the dataset for a training job, for example, <b>/usr/data/</b> |

**Table 8-29** remote\_constraint

| Parameter  | Type   | Description                                                                                                                                                                                                                                                |
|------------|--------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| data_type  | String | Data input type, including the data storage location and dataset                                                                                                                                                                                           |
| attributes | String | Attributes when a dataset functions as the data input<br>Options: <ul style="list-style-type: none"> <li>• <b>data_format</b>: data format</li> <li>• <b>data_segmentation</b>: data segmentation</li> <li>• <b>dataset_type</b>: data labeling</li> </ul> |

**Table 8-30** Output

| Parameter   | Type                 | Description                                                                 |
|-------------|----------------------|-----------------------------------------------------------------------------|
| name        | String               | Name of the data output channel                                             |
| description | String               | Description of the data output channel                                      |
| local_dir   | String               | Local directory of the container to which the data output channel is mapped |
| remote      | <b>remote</b> object | Information of the data output                                              |

**Table 8-31** remote

| Parameter | Type              | Description                   |
|-----------|-------------------|-------------------------------|
| obs       | <b>obs</b> object | OBS to which data is exported |

**Table 8-32** obs

| Parameter | Type   | Description                       |
|-----------|--------|-----------------------------------|
| obs_url   | String | OBS URL to which data is exported |

**Table 8-33** engine

| Parameter      | Type   | Description                                                                                                                               |
|----------------|--------|-------------------------------------------------------------------------------------------------------------------------------------------|
| engine_id      | String | Engine ID selected for a training job, which can be <b>engine_id</b> , <b>engine_name</b> and <b>engine_version</b> , or <b>image_url</b> |
| engine_name    | String | Name of the engine selected for a training job. Leave this parameter blank if <b>engine_id</b> is specified.                              |
| engine_version | String | Version of the engine selected for a training job. Leave this parameter blank if <b>engine_id</b> is specified.                           |
| image_url      | String | Custom image URL selected for a training job                                                                                              |



**Table 8-34** TaskResponse

| Parameter     | Type                         | Description                                                                                                                                                                                         |
|---------------|------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| role          | String                       | Role of a heterogeneous training job task<br>Options: <ul style="list-style-type: none"> <li>• <b>learner</b>: GPUs or CPUs are supported.</li> <li>• <b>worker</b>: CPUs are supported.</li> </ul> |
| algorithm     | <b>algorithm</b> object      | Algorithm configurations in algorithm management                                                                                                                                                    |
| task_resource | <b>FlavorResponse</b> object | Flavors for a training job or an algorithm                                                                                                                                                          |

**Table 8-35** algorithm

| Parameter | Type                  | Description                                                            |
|-----------|-----------------------|------------------------------------------------------------------------|
| code_dir  | String                | Absolute path of the directory where the algorithm boot file is stored |
| boot_file | String                | Absolute path of the algorithm boot file                               |
| inputs    | <b>inputs</b> object  | Algorithm input channel                                                |
| outputs   | <b>outputs</b> object | Algorithm output channel                                               |
| engine    | <b>engine</b> object  | Engine on which a heterogeneous job depends                            |

**Table 8-36** inputs

| Parameter | Type                 | Description                                                                        |
|-----------|----------------------|------------------------------------------------------------------------------------|
| name      | String               | Name of the data input channel                                                     |
| local_dir | String               | Local path of the container to which the data input and output channels are mapped |
| remote    | <b>remote</b> object | Actual data input, which can only be OBS for heterogeneous jobs                    |

**Table 8-37** remote

| Parameter | Type                 | Description                                   |
|-----------|----------------------|-----------------------------------------------|
| obs       | <b>obs</b><br>object | OBS in which data input and output are stored |

**Table 8-38** obs

| Parameter | Type   | Description                                                               |
|-----------|--------|---------------------------------------------------------------------------|
| obs_url   | String | OBS URL of the dataset for a training job, for example, <b>/usr/data/</b> |

**Table 8-39** outputs

| Parameter | Type                    | Description                                                                 |
|-----------|-------------------------|-----------------------------------------------------------------------------|
| name      | String                  | Name of the data output channel                                             |
| local_dir | String                  | Local directory of the container to which the data output channel is mapped |
| remote    | <b>remote</b><br>object | Information of the data output                                              |
| mode      | String                  | Data transmission mode, which defaults to <b>upload_periodically</b>        |
| period    | String                  | Data transmission period, which defaults to <b>30s</b>                      |

**Table 8-40** remote

| Parameter | Type                 | Description                   |
|-----------|----------------------|-------------------------------|
| obs       | <b>obs</b><br>object | OBS to which data is exported |

**Table 8-41** obs

| Parameter | Type   | Description                       |
|-----------|--------|-----------------------------------|
| obs_url   | String | OBS URL to which data is exported |

**Table 8-42** engine

| Parameter      | Type    | Description                                                                 |
|----------------|---------|-----------------------------------------------------------------------------|
| engine_id      | String  | Engine ID of a heterogeneous job, for example, <b>caffe-1.0.0-python2.7</b> |
| engine_name    | String  | Engine name of a heterogeneous job, for example, <b>Caffe</b>               |
| engine_version | String  | Engine version of a heterogeneous job                                       |
| v1_compatible  | Boolean | Whether v1 is compatible                                                    |
| run_user       | String  | User UID for which the engine is started by default                         |

**Table 8-43** FlavorResponse

| Parameter   | Type                      | Description                                                                                                                                  |
|-------------|---------------------------|----------------------------------------------------------------------------------------------------------------------------------------------|
| flavor_id   | String                    | ID of the resource flavor                                                                                                                    |
| flavor_name | String                    | Name of the resource flavor                                                                                                                  |
| max_num     | Integer                   | Maximum number of nodes with the resource flavor                                                                                             |
| flavor_type | String                    | Resource flavor type. Options: <ul style="list-style-type: none"> <li>• <b>CPU</b></li> <li>• <b>GPU</b></li> <li>• <b>Ascend</b></li> </ul> |
| billing     | <b>billing</b> object     | Billing information of a resource flavor                                                                                                     |
| flavor_info | <b>flavor_info</b> object | Resource flavor details                                                                                                                      |
| attributes  | Map<String,String>        | Other flavor attributes                                                                                                                      |

**Table 8-44** billing

| Parameter   | Type    | Description             |
|-------------|---------|-------------------------|
| code        | String  | Billing code            |
| unit_number | Integer | Number of billing units |

**Table 8-45** flavor\_info

| Parameter  | Type                 | Description                                                                                                        |
|------------|----------------------|--------------------------------------------------------------------------------------------------------------------|
| max_number | Integer              | Maximum number of nodes that can be selected. Value <b>1</b> indicates that the distributed mode is not supported. |
| cpu        | <b>cpu</b> object    | CPU specifications                                                                                                 |
| gpu        | <b>gpu</b> object    | GPU specifications                                                                                                 |
| npu        | <b>npu</b> object    | Ascend specifications                                                                                              |
| memory     | <b>memory</b> object | Memory information                                                                                                 |

**Table 8-46** cpu

| Parameter   | Type    | Description      |
|-------------|---------|------------------|
| arch        | String  | CPU architecture |
| core_number | Integer | Number of cores  |

**Table 8-47** gpu

| Parameter    | Type    | Description    |
|--------------|---------|----------------|
| unit_number  | Integer | Number of GPUs |
| product_name | String  | Product name   |

| Parameter | Type   | Description |
|-----------|--------|-------------|
| memory    | String | Memory      |

**Table 8-48** npu

| Parameter    | Type   | Description    |
|--------------|--------|----------------|
| unit_num     | String | Number of NPUs |
| product_name | String | Product name   |
| memory       | String | Memory         |

**Table 8-49** memory

| Parameter | Type    | Description            |
|-----------|---------|------------------------|
| size      | Integer | Memory size            |
| unit      | String  | Number of memory units |

**Table 8-50** spec

| Parameter       | Type                          | Description                                                                                                     |
|-----------------|-------------------------------|-----------------------------------------------------------------------------------------------------------------|
| resource        | <b>Resource</b> object        | Resource flavors of a training job, which can either be <b>flavor_id</b> or <b>pool_id</b> and <b>flavor_id</b> |
| volumes         | Array of objects              | Volumes attached for a training job                                                                             |
| log_export_path | <b>log_export_path</b> object | Export path of training job logs                                                                                |

**Table 8-51** Resource

| Parameter     | Type                        | Description                                                                                          |
|---------------|-----------------------------|------------------------------------------------------------------------------------------------------|
| policy        | String                      | Resource flavor mode of a training job. Options: <b>regular</b> , <b>economic</b> , and <b>turbo</b> |
| flavor_id     | String                      | Resource flavor ID of a training job                                                                 |
| flavor_name   | String                      | Read-only flavor name returned by ModelArts when <b>flavor_id</b> is specified                       |
| node_count    | Integer                     | Number of resource replicas selected for a training job<br>Minimum value: 1                          |
| pool_id       | String                      | Resource pool ID selected for a training job                                                         |
| flavor_detail | <b>flavor_detail</b> object | Flavors for a training job or an algorithm                                                           |

**Table 8-52** flavor\_detail

| Parameter   | Type                      | Description                                                                                                             |
|-------------|---------------------------|-------------------------------------------------------------------------------------------------------------------------|
| flavor_type | String                    | Resource flavor type. Options: <ul style="list-style-type: none"> <li>• CPU</li> <li>• GPU</li> <li>• Ascend</li> </ul> |
| billing     | <b>billing</b> object     | Billing information of a resource flavor                                                                                |
| flavor_info | <b>flavor_info</b> object | Resource flavor details                                                                                                 |

**Table 8-53** billing

| Parameter | Type    | Description             |
|-----------|---------|-------------------------|
| code      | String  | Billing code            |
| unit_num  | Integer | Number of billing units |

**Table 8-54** flavor\_info

| Parameter | Type                 | Description                                                                                                        |
|-----------|----------------------|--------------------------------------------------------------------------------------------------------------------|
| max_num   | Integer              | Maximum number of nodes that can be selected. Value <b>1</b> indicates that the distributed mode is not supported. |
| cpu       | <b>cpu</b> object    | CPU specifications                                                                                                 |
| gpu       | <b>gpu</b> object    | GPU specifications                                                                                                 |
| npu       | <b>npu</b> object    | Ascend specifications                                                                                              |
| memory    | <b>memory</b> object | Memory information                                                                                                 |
| disk      | <b>disk</b> object   | Disk information                                                                                                   |

**Table 8-55** cpu

| Parameter | Type    | Description      |
|-----------|---------|------------------|
| arch      | String  | CPU architecture |
| core_num  | Integer | Number of cores  |

**Table 8-56** gpu

| Parameter    | Type    | Description    |
|--------------|---------|----------------|
| unit_num     | Integer | Number of GPUs |
| product_name | String  | Product name   |
| memory       | String  | Memory         |

**Table 8-57** npu

| Parameter    | Type   | Description    |
|--------------|--------|----------------|
| unit_num     | String | Number of NPUs |
| product_name | String | Product name   |
| memory       | String | Memory         |

**Table 8-58** memory

| Parameter | Type    | Description            |
|-----------|---------|------------------------|
| size      | Integer | Memory size            |
| unit      | String  | Number of memory units |

**Table 8-59** disk

| Parameter | Type   | Description                                  |
|-----------|--------|----------------------------------------------|
| size      | String | Disk size                                    |
| unit      | String | Unit of the disk size, which is GB generally |

**Table 8-60** volumes

| Parameter | Type                 | Description                |
|-----------|----------------------|----------------------------|
| nfs       | <b>nfs</b><br>object | Disks attached in NFS mode |

**Table 8-61** nfs

| Parameter       | Type   | Description     |
|-----------------|--------|-----------------|
| nfs_server_path | String | NFS server path |



| Parameter  | Type    | Description                                                           |
|------------|---------|-----------------------------------------------------------------------|
| local_path | String  | Path for attaching disks to the training container                    |
| read_only  | Boolean | Whether the disks attached to the container in NFS mode are read-only |

**Table 8-62** log\_export\_path

| Parameter | Type   | Description                                         |
|-----------|--------|-----------------------------------------------------|
| obs_url   | String | OBS URL for storing training job logs               |
| host_path | String | Path of the host where training job logs are stored |

**Table 8-63** Response for the failure to call a training API

| Parameter      | Type   | Description                                                                                                                                                               |
|----------------|--------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| error_msg      | String | Error message when calling an API failed. This parameter is unavailable if an API is successfully called.                                                                 |
| error_code     | String | Error code when calling an API failed. For details, see "Error Codes" in <i>ModelArts API Reference</i> . This parameter is unavailable if an API is successfully called. |
| error_solution | String | Solution to an API calling failure. This parameter is unavailable if an API is successfully called.                                                                       |

**Table 8-64** Response for the failure to call a training API

| Parameter      | Type   | Description                                                                                                                                                               |
|----------------|--------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| error_msg      | String | Error message when calling an API failed. This parameter is unavailable if an API is successfully called.                                                                 |
| error_code     | String | Error code when calling an API failed. For details, see "Error Codes" in <i>ModelArts API Reference</i> . This parameter is unavailable if an API is successfully called. |
| error_solution | String | Solution to an API calling failure. This parameter is unavailable if an API is successfully called.                                                                       |

## 8.1.4 Obtaining the Details About a Training Job

### Sample Code

In ModelArts notebook, you do not need to enter authentication parameters for session authentication. For details about session authentication of other development environments, see [Session Authentication](#).

- Method 1: Use the specified **job\_id**.  

```
from modelarts.session import Session
from modelarts.estimatorV2 import Estimator
session = Session()
estimator = Estimator(session=session, job_id="618222c4-dc2f-4cfe-bc49-72b075b7552f")
job_info = estimator.get_job_info()
print(job_info)
```
- Method 2: Use the training job created in [Creating a Training Job](#).  

```
job_info = job_instance.get_job_info()
print(job_info)
```

### Parameters

**Table 8-65** Estimator request parameters

| Parameter | Mandatory | Type   | Description                                                                                                                                                                                                                                         |
|-----------|-----------|--------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| session   | Yes       | Object | Session object. For details about the initialization method, see <a href="#">Session Authentication</a> .                                                                                                                                           |
| job_id    | Yes       | String | ID of a training job. You can obtain <b>job_id</b> using the training job created in <a href="#">Creating a Training Job</a> , for example, <b>job_instance.job_id</b> , or from the response obtained in <a href="#">Obtaining Training Jobs</a> . |

**Table 8-66** `get_job_info` response parameters

| Parameter | Type                               | Description                                                                                                                                                                                                                                                                                                          |
|-----------|------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| kind      | String                             | Training job type, which defaults to <b>job</b> .<br>Options: <ul style="list-style-type: none"> <li>• <b>job</b>: training job</li> <li>• <b>hetero_job</b>: heterogeneous job</li> <li>• <b>autosearch_job</b>: auto search job</li> <li>• <b>mrs_job</b>: MRS job</li> <li>• <b>edge_job</b>: edge job</li> </ul> |
| metadata  | <a href="#">JobMetadata</a> object | Metadata of a training job.                                                                                                                                                                                                                                                                                          |

| Parameter | Type                                 | Description                                                                                                                                                                                                                                                                                                                                                                                                      |
|-----------|--------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| status    | <b>Status</b> object                 | Status of a training job. When creating a training job, you do not need to set this parameter.                                                                                                                                                                                                                                                                                                                   |
| algorithm | <b>JobAlgorithmResponse</b> object   | Algorithm used by a training job. The following formats are supported: <ul style="list-style-type: none"> <li>• <b>id</b>: Only the algorithm ID is used.</li> <li>• <b>subscription_id</b> and <b>item_version_id</b>: The subscription ID and version ID of the algorithm are used.</li> <li>• <b>code_dir</b> and <b>boot_file</b>: The code directory and boot file of the training job are used.</li> </ul> |
| tasks     | Array of <b>TaskResponse</b> objects | Tasks of a heterogeneous training job.                                                                                                                                                                                                                                                                                                                                                                           |
| spec      | <b>spec</b> object                   | Specifications of a training job.                                                                                                                                                                                                                                                                                                                                                                                |

**Table 8-67** JobMetadata

| Parameter    | Type                | Description                                                                                                                                                                                |
|--------------|---------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| id           | String              | Training job ID, which is generated and returned by ModelArts after a training job is created.                                                                                             |
| name         | String              | Name of a training job. The value must contain 1 to 64 characters consisting of only digits, letters, underscores (_), and hyphens (-).                                                    |
| workspace_id | String              | Workspace where a training job is deployed. Default value: <b>0</b>                                                                                                                        |
| description  | String              | Description of a training job, which defaults to <b>NULL</b> . The value must contain 0 to 256 characters.                                                                                 |
| create_time  | Long                | Time when a training job was created, in milliseconds. The value is generated and returned by ModelArts after a training job is created.                                                   |
| username     | String              | Username for creating a training job. The username is generated and returned by ModelArts after a training job is created.                                                                 |
| annotations  | Map<String, String> | Declaration template of a training job. For heterogeneous jobs, the default value of <b>job_template</b> is <b>Template RL</b> . For other jobs, the default value is <b>Template DL</b> . |

**Table 8-68** Status

| Parameter          | Type                  | Description                                                                                                                                                                                                  |
|--------------------|-----------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| phase              | String                | Level-1 status of a training job. The value will remain unchanged. Options: <b>Creating, Pending, Running, Failed, Completed, Terminating, Terminated,</b> and <b>Abnormal</b>                               |
| secondary_phase    | String                | Level-2 status of a training job. The value can be changed. Options: <b>Creating, Queuing, Running, Failed, Completed, Terminating, Terminated, CreateFailed, TerminatedFailed, Unknown,</b> and <b>Lost</b> |
| duration           | Long                  | Running duration of a training job, in milliseconds                                                                                                                                                          |
| node_count_metrics | Array<Array<Integer>> | Node count changes during the runtime of a training job                                                                                                                                                      |
| tasks              | Array of strings      | Task of a training job                                                                                                                                                                                       |
| start_time         | String                | Start time of a training job. The value is in timestamp format.                                                                                                                                              |
| task_statuses      | Array of objects      | Status of a training job task                                                                                                                                                                                |

**Table 8-69** task\_statuses

| Parameter | Type    | Description                          |
|-----------|---------|--------------------------------------|
| task      | String  | Task of a training job               |
| exit_code | Integer | Exit code of a training job task     |
| message   | String  | Error message of a training job task |

**Table 8-70** JobAlgorithmResponse

| Parameter | Type   | Description                                                                                                                                                                                                                                                                                                                                                        |
|-----------|--------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| id        | String | Algorithm ID<br>Options: <ul style="list-style-type: none"> <li>• <b>id</b>: Only the algorithm ID is used.</li> <li>• <b>subscription_id</b> and <b>item_version_id</b>: The subscription ID and version ID of the algorithm are used.</li> <li>• <b>code_dir</b> and <b>boot_file</b>: The code directory and boot file of the training job are used.</li> </ul> |

| Parameter                 | Type                              | Description                                                                                                                                                                                                                                                                      |
|---------------------------|-----------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| name                      | String                            | Algorithm name                                                                                                                                                                                                                                                                   |
| subscription_id           | String                            | Subscription ID of the subscribed algorithm, which must be used with <b>item_version_id</b>                                                                                                                                                                                      |
| item_version_id           | String                            | Version ID of the subscribed algorithm, which must be used with <b>subscription_id</b>                                                                                                                                                                                           |
| code_dir                  | String                            | Code directory of a training job, for example, <b>/usr/app/</b> . This parameter must be used with <b>boot_file</b> . Leave this parameter blank if <b>id</b> , or <b>subscription_id</b> and <b>item_version_id</b> are specified.                                              |
| boot_file                 | String                            | Boot file of a training job, which must be stored in the code directory, for example, <b>/usr/app/boot.py</b> . This parameter must be used with <b>code_dir</b> . Leave this parameter blank if <b>id</b> , or <b>subscription_id</b> and <b>item_version_id</b> are specified. |
| autosearch_config_path    | String                            | YAML configuration path of an auto search job. An OBS URL is required.                                                                                                                                                                                                           |
| autosearch_framework_path | String                            | Framework code directory of an auto search job. An OBS URL is required.                                                                                                                                                                                                          |
| command                   | String                            | Boot command for starting the container of the custom image used for creating a training job. The value of this parameter can be the same as the <b>code_dir</b> value.                                                                                                          |
| parameters                | Array of <b>Parameter</b> objects | Running parameters of a training job.                                                                                                                                                                                                                                            |
| policies                  | <b>policies</b> object            | Policies supported by a training job.                                                                                                                                                                                                                                            |
| inputs                    | Array of <b>Input</b> objects     | Input of a training job.                                                                                                                                                                                                                                                         |
| outputs                   | Array of <b>Output</b> objects    | Output of a training job.                                                                                                                                                                                                                                                        |
| engine                    | <b>engine</b> object              | Engine of a training job. Leave this parameter blank if the job is created using <b>id</b> of the algorithm in algorithm management, or <b>subscription_id</b> and <b>item_version_id</b> of the subscribed algorithm.                                                           |

| Parameter    | Type               | Description                                                                                         |
|--------------|--------------------|-----------------------------------------------------------------------------------------------------|
| environments | Map<String,String> | Environment variables of a training job in the format of "key":"value". Leave this parameter blank. |

**Table 8-71** Parameter

| Parameter        | Type                           | Description                      |
|------------------|--------------------------------|----------------------------------|
| name             | String                         | Parameter name                   |
| value            | String                         | Parameter value                  |
| description      | String                         | Parameter description            |
| constraint       | <b>constraint</b> object       | Parameter constraint             |
| i18n_description | <b>i18n_description</b> object | Internationalization description |

**Table 8-72** constraint

| Parameter   | Type             | Description                        |
|-------------|------------------|------------------------------------|
| type        | String           | Parameter type                     |
| editable    | Boolean          | Whether the parameter is editable  |
| required    | Boolean          | Whether the parameter is mandatory |
| sensitive   | Boolean          | Whether the parameter is sensitive |
| valid_type  | String           | Valid type                         |
| valid_range | Array of strings | Valid range                        |

**Table 8-73** i18n\_description

| Parameter   | Type   | Description                   |
|-------------|--------|-------------------------------|
| language    | String | Internationalization language |
| description | String | Description                   |

**Table 8-74** policies

| Parameter   | Type                               | Description                         |
|-------------|------------------------------------|-------------------------------------|
| auto_search | <a href="#">auto_search</a> object | Hyperparameter search configuration |

**Table 8-75** auto\_search

| Parameter          | Type             | Description                                       |
|--------------------|------------------|---------------------------------------------------|
| skip_search_params | String           | Hyperparameter parameters that need to be skipped |
| reward_attrs       | Array of objects | Search metrics                                    |
| search_params      | Array of objects | Search parameters                                 |
| algo_configs       | Array of objects | Search algorithm configurations                   |

**Table 8-76** reward\_attrs

| Parameter | Type   | Description                                                                                                                                                                 |
|-----------|--------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| name      | String | Metric name                                                                                                                                                                 |
| mode      | String | Search mode <ul style="list-style-type: none"> <li>• <b>max</b>: A larger metric value is preferred.</li> <li>• <b>min</b>: A smaller metric value is preferred.</li> </ul> |
| regex     | String | Regular expression of a metric                                                                                                                                              |

**Table 8-77** search\_params

| Parameter   | Type   | Description                                                                                                                                                                       |
|-------------|--------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| name        | String | Hyperparameter name                                                                                                                                                               |
| param_type  | String | Parameter type <ul style="list-style-type: none"> <li>• <b>continuous</b>: Parameter values are continuous.</li> <li>• <b>discrete</b>: Parameter values are discrete.</li> </ul> |
| lower_bound | String | Lower bound of the hyperparameter                                                                                                                                                 |
| upper_bound | String | Upper bound of the hyperparameter                                                                                                                                                 |

| Parameter           | Type             | Description                                                          |
|---------------------|------------------|----------------------------------------------------------------------|
| discrete_points_num | String           | Number of discrete points of a hyperparameter with continuous values |
| discrete_values     | Array of strings | Discrete hyperparameter values                                       |

**Table 8-78** algo\_configs

| Parameter | Type                                                           | Description                  |
|-----------|----------------------------------------------------------------|------------------------------|
| name      | String                                                         | Name of the search algorithm |
| params    | Array of <a href="#">AutoSearchAlgoConfigParameter</a> objects | Search algorithm parameters  |

**Table 8-79** AutoSearchAlgoConfigParameter

| Parameter | Type   | Description     |
|-----------|--------|-----------------|
| key       | String | Parameter key   |
| value     | String | Parameter value |
| type      | String | Parameter type  |

**Table 8-80** Input

| Parameter         | Type                                 | Description                                                                |
|-------------------|--------------------------------------|----------------------------------------------------------------------------|
| name              | String                               | Name of the data input channel                                             |
| description       | String                               | Description of the data input channel                                      |
| local_dir         | String                               | Local directory of the container to which the data input channel is mapped |
| remote            | <a href="#">InputDataInfo</a> object | Information of the data input                                              |
| remote_constraint | Array of objects                     | Data input constraint                                                      |



**Table 8-81** InputDataInfo

| Parameter | Type                  | Description                                   |
|-----------|-----------------------|-----------------------------------------------|
| dataset   | <b>dataset</b> object | Dataset as the data input                     |
| obs       | <b>obs</b> object     | OBS in which data input and output are stored |

**Table 8-82** dataset

| Parameter  | Type   | Description                                                                                                                                                           |
|------------|--------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| id         | String | Dataset ID of a training job                                                                                                                                          |
| version_id | String | Dataset version ID of a training job                                                                                                                                  |
| obs_url    | String | OBS URL of the dataset for a training job, which is automatically parsed by ModelArts based on the dataset ID and dataset version IDs, for example, <b>/usr/data/</b> |

**Table 8-83** obs

| Parameter | Type   | Description                                                               |
|-----------|--------|---------------------------------------------------------------------------|
| obs_url   | String | OBS URL of the dataset for a training job, for example, <b>/usr/data/</b> |

**Table 8-84** remote\_constraint

| Parameter  | Type   | Description                                                                                                                                                                                                                                                |
|------------|--------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| data_type  | String | Data input type, including the data storage location and dataset                                                                                                                                                                                           |
| attributes | String | Attributes when a dataset functions as the data input<br>Options: <ul style="list-style-type: none"> <li>• <b>data_format</b>: data format</li> <li>• <b>data_segmentation</b>: data segmentation</li> <li>• <b>dataset_type</b>: data labeling</li> </ul> |

**Table 8-85** Output

| Parameter   | Type                 | Description                                                                 |
|-------------|----------------------|-----------------------------------------------------------------------------|
| name        | String               | Name of the data output channel                                             |
| description | String               | Description of the data output channel                                      |
| local_dir   | String               | Local directory of the container to which the data output channel is mapped |
| remote      | <b>remote</b> object | Information of the data output                                              |

**Table 8-86** remote

| Parameter | Type              | Description                   |
|-----------|-------------------|-------------------------------|
| obs       | <b>obs</b> object | OBS to which data is exported |

**Table 8-87** obs

| Parameter | Type   | Description                       |
|-----------|--------|-----------------------------------|
| obs_url   | String | OBS URL to which data is exported |

**Table 8-88** engine

| Parameter      | Type   | Description                                                                                                                               |
|----------------|--------|-------------------------------------------------------------------------------------------------------------------------------------------|
| engine_id      | String | Engine ID selected for a training job, which can be <b>engine_id</b> , <b>engine_name</b> and <b>engine_version</b> , or <b>image_url</b> |
| engine_name    | String | Name of the engine selected for a training job. Leave this parameter blank if <b>engine_id</b> is specified.                              |
| engine_version | String | Version of the engine selected for a training job. Leave this parameter blank if <b>engine_id</b> is specified.                           |
| image_url      | String | Custom image URL selected for a training job                                                                                              |

**Table 8-89** TaskResponse

| Parameter     | Type                         | Description                                                                                                                                                                                         |
|---------------|------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| role          | String                       | Role of a heterogeneous training job task<br>Options: <ul style="list-style-type: none"> <li>• <b>learner</b>: GPUs or CPUs are supported.</li> <li>• <b>worker</b>: CPUs are supported.</li> </ul> |
| algorithm     | <b>algorithm</b> object      | Algorithm configurations in algorithm management                                                                                                                                                    |
| task_resource | <b>FlavorResponse</b> object | Flavors for a training job or an algorithm                                                                                                                                                          |

**Table 8-90** algorithm

| Parameter | Type                  | Description                                                            |
|-----------|-----------------------|------------------------------------------------------------------------|
| code_dir  | String                | Absolute path of the directory where the algorithm boot file is stored |
| boot_file | String                | Absolute path of the algorithm boot file                               |
| inputs    | <b>inputs</b> object  | Algorithm input channel                                                |
| outputs   | <b>outputs</b> object | Algorithm output channel                                               |
| engine    | <b>engine</b> object  | Engine on which a heterogeneous job depends                            |

**Table 8-91** inputs

| Parameter | Type                 | Description                                                                        |
|-----------|----------------------|------------------------------------------------------------------------------------|
| name      | String               | Name of the data input channel                                                     |
| local_dir | String               | Local path of the container to which the data input and output channels are mapped |
| remote    | <b>remote</b> object | Actual data input, which can only be OBS for heterogeneous jobs                    |

**Table 8-92** remote

| Parameter | Type                 | Description                                   |
|-----------|----------------------|-----------------------------------------------|
| obs       | <b>obs</b><br>object | OBS in which data input and output are stored |

**Table 8-93** obs

| Parameter | Type   | Description                                                               |
|-----------|--------|---------------------------------------------------------------------------|
| obs_url   | String | OBS URL of the dataset for a training job, for example, <b>/usr/data/</b> |

**Table 8-94** outputs

| Parameter | Type                    | Description                                                                 |
|-----------|-------------------------|-----------------------------------------------------------------------------|
| name      | String                  | Name of the data output channel                                             |
| local_dir | String                  | Local directory of the container to which the data output channel is mapped |
| remote    | <b>remote</b><br>object | Information of the data output                                              |
| mode      | String                  | Data transmission mode, which defaults to <b>upload_periodically</b>        |
| period    | String                  | Data transmission period, which defaults to <b>30s</b>                      |

**Table 8-95** remote

| Parameter | Type                 | Description                   |
|-----------|----------------------|-------------------------------|
| obs       | <b>obs</b><br>object | OBS to which data is exported |

**Table 8-96** obs

| Parameter | Type   | Description                       |
|-----------|--------|-----------------------------------|
| obs_url   | String | OBS URL to which data is exported |

**Table 8-97** engine

| Parameter      | Type    | Description                                                                 |
|----------------|---------|-----------------------------------------------------------------------------|
| engine_id      | String  | Engine ID of a heterogeneous job, for example, <b>caffe-1.0.0-python2.7</b> |
| engine_name    | String  | Engine name of a heterogeneous job, for example, <b>Caffe</b>               |
| engine_version | String  | Engine version of a heterogeneous job                                       |
| v1_compatible  | Boolean | Whether v1 is compatible                                                    |
| run_user       | String  | User UID for which the engine is started by default                         |

**Table 8-98** FlavorResponse

| Parameter   | Type                      | Description                                                                                                             |
|-------------|---------------------------|-------------------------------------------------------------------------------------------------------------------------|
| flavor_id   | String                    | ID of the resource flavor                                                                                               |
| flavor_name | String                    | Name of the resource flavor                                                                                             |
| max_num     | Integer                   | Maximum number of nodes with the resource flavor                                                                        |
| flavor_type | String                    | Resource flavor type. Options: <ul style="list-style-type: none"> <li>• CPU</li> <li>• GPU</li> <li>• Ascend</li> </ul> |
| billing     | <b>billing</b> object     | Billing information of a resource flavor                                                                                |
| flavor_info | <b>flavor_info</b> object | Resource flavor details                                                                                                 |
| attributes  | Map<String,String>        | Other flavor attributes                                                                                                 |

**Table 8-99** billing

| Parameter | Type    | Description             |
|-----------|---------|-------------------------|
| code      | String  | Billing code            |
| unit_num  | Integer | Number of billing units |

**Table 8-100** flavor\_info

| Parameter | Type                 | Description                                                                                                        |
|-----------|----------------------|--------------------------------------------------------------------------------------------------------------------|
| max_num   | Integer              | Maximum number of nodes that can be selected. Value <b>1</b> indicates that the distributed mode is not supported. |
| cpu       | <b>cpu</b> object    | CPU specifications                                                                                                 |
| gpu       | <b>gpu</b> object    | GPU specifications                                                                                                 |
| npu       | <b>npu</b> object    | Ascend specifications                                                                                              |
| memory    | <b>memory</b> object | Memory information                                                                                                 |

**Table 8-101** cpu

| Parameter | Type    | Description      |
|-----------|---------|------------------|
| arch      | String  | CPU architecture |
| core_num  | Integer | Number of cores  |

**Table 8-102** gpu

| Parameter    | Type    | Description    |
|--------------|---------|----------------|
| unit_num     | Integer | Number of GPUs |
| product_name | String  | Product name   |
| memory       | String  | Memory         |

**Table 8-103** npu

| Parameter    | Type   | Description    |
|--------------|--------|----------------|
| unit_num     | String | Number of NPUs |
| product_name | String | Product name   |
| memory       | String | Memory         |

**Table 8-104** memory

| Parameter | Type    | Description            |
|-----------|---------|------------------------|
| size      | Integer | Memory size            |
| unit      | String  | Number of memory units |

**Table 8-105** spec

| Parameter       | Type                                   | Description                                                                                                     |
|-----------------|----------------------------------------|-----------------------------------------------------------------------------------------------------------------|
| resource        | <a href="#">Resource</a> object        | Resource flavors of a training job, which can either be <b>flavor_id</b> or <b>pool_id</b> and <b>flavor_id</b> |
| volumes         | Array of objects                       | Volumes attached for a training job                                                                             |
| log_export_path | <a href="#">log_export_path</a> object | Export path of training job logs                                                                                |

**Table 8-106** Resource

| Parameter     | Type                                 | Description                                                                                          |
|---------------|--------------------------------------|------------------------------------------------------------------------------------------------------|
| policy        | String                               | Resource flavor mode of a training job. Options: <b>regular</b> , <b>economic</b> , and <b>turbo</b> |
| flavor_id     | String                               | Resource flavor ID of a training job                                                                 |
| flavor_name   | String                               | Read-only flavor name returned by ModelArts when <b>flavor_id</b> is specified                       |
| node_count    | Integer                              | Number of resource replicas selected for a training job<br>Minimum value: <b>1</b>                   |
| pool_id       | String                               | Resource pool ID selected for a training job                                                         |
| flavor_detail | <a href="#">flavor_detail</a> object | Flavors for a training job or an algorithm                                                           |

**Table 8-107** flavor\_detail

| Parameter   | Type                      | Description                                                                                                             |
|-------------|---------------------------|-------------------------------------------------------------------------------------------------------------------------|
| flavor_type | String                    | Resource flavor type. Options: <ul style="list-style-type: none"> <li>• CPU</li> <li>• GPU</li> <li>• Ascend</li> </ul> |
| billing     | <b>billing</b> object     | Billing information of a resource flavor                                                                                |
| flavor_info | <b>flavor_info</b> object | Resource flavor details                                                                                                 |

**Table 8-108** billing

| Parameter | Type    | Description             |
|-----------|---------|-------------------------|
| code      | String  | Billing code            |
| unit_num  | Integer | Number of billing units |

**Table 8-109** flavor\_info

| Parameter | Type                 | Description                                                                                                        |
|-----------|----------------------|--------------------------------------------------------------------------------------------------------------------|
| max_num   | Integer              | Maximum number of nodes that can be selected. Value <b>1</b> indicates that the distributed mode is not supported. |
| cpu       | <b>cpu</b> object    | CPU specifications                                                                                                 |
| gpu       | <b>gpu</b> object    | GPU specifications                                                                                                 |
| npu       | <b>npu</b> object    | Ascend specifications                                                                                              |
| memory    | <b>memory</b> object | Memory information                                                                                                 |
| disk      | <b>disk</b> object   | Disk information                                                                                                   |



**Table 8-110** cpu

| Parameter | Type    | Description      |
|-----------|---------|------------------|
| arch      | String  | CPU architecture |
| core_num  | Integer | Number of cores  |

**Table 8-111** gpu

| Parameter    | Type    | Description    |
|--------------|---------|----------------|
| unit_num     | Integer | Number of GPUs |
| product_name | String  | Product name   |
| memory       | String  | Memory         |

**Table 8-112** npu

| Parameter    | Type   | Description    |
|--------------|--------|----------------|
| unit_num     | String | Number of NPUs |
| product_name | String | Product name   |
| memory       | String | Memory         |

**Table 8-113** memory

| Parameter | Type    | Description            |
|-----------|---------|------------------------|
| size      | Integer | Memory size            |
| unit      | String  | Number of memory units |

**Table 8-114** disk

| Parameter | Type   | Description                                  |
|-----------|--------|----------------------------------------------|
| size      | String | Disk size                                    |
| unit      | String | Unit of the disk size, which is GB generally |

**Table 8-115** volumes

| Parameter | Type       | Description                |
|-----------|------------|----------------------------|
| nfs       | nfs object | Disks attached in NFS mode |

**Table 8-116** nfs

| Parameter       | Type    | Description                                                           |
|-----------------|---------|-----------------------------------------------------------------------|
| nfs_server_path | String  | NFS server path                                                       |
| local_path      | String  | Path for attaching disks to the training container                    |
| read_only       | Boolean | Whether the disks attached to the container in NFS mode are read-only |

**Table 8-117** log\_export\_path

| Parameter | Type   | Description                                         |
|-----------|--------|-----------------------------------------------------|
| obs_url   | String | OBS URL for storing training job logs               |
| host_path | String | Path of the host where training job logs are stored |

**Table 8-118** Response for the failure to call a training API

| Parameter      | Type   | Description                                                                                                                                                               |
|----------------|--------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| error_msg      | String | Error message when calling an API failed. This parameter is unavailable if an API is successfully called.                                                                 |
| error_code     | String | Error code when calling an API failed. For details, see "Error Codes" in <i>ModelArts API Reference</i> . This parameter is unavailable if an API is successfully called. |
| error_solution | String | Solution to an API calling failure. This parameter is unavailable if an API is successfully called.                                                                       |

## 8.1.5 Modifying the Description of a Training Job

### Sample Code

In ModelArts notebook, you do not need to enter authentication parameters for session authentication. For details about session authentication of other development environments, see [Session Authentication](#).

- Method 1: Use the specified **job\_id**.  

```
from modelarts.session import Session
from modelarts.estimatorV2 import Estimator
session = Session()
estimator = Estimator(session=session, job_id="your job id")
estimator.update_job_configs(description="update job description")
```
- Method 2: Use the training job created in [Creating a Training Job](#).  

```
job_instance.update_job_configs(description="update job description fourth")
```

### Parameters

**Table 8-119** Estimator request parameters

| Parameter | Mandatory | Type   | Description                                                                                                                                                                                                                                         |
|-----------|-----------|--------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| session   | Yes       | Object | Session object. For details about the initialization method, see <a href="#">Session Authentication</a> .                                                                                                                                           |
| job_id    | Yes       | String | ID of a training job. You can obtain <b>job_id</b> using the training job created in <a href="#">Creating a Training Job</a> , for example, <b>job_instance.job_id</b> , or from the response obtained in <a href="#">Obtaining Training Jobs</a> . |

**Table 8-120** update\_job\_configs request parameters

| Parameter   | Mandatory | Type   | Description                                    |
|-------------|-----------|--------|------------------------------------------------|
| description | Yes       | String | Description of the training job to be modified |

There is no response for successfully calling an API.

**Table 8-121** Response for the failure to call a training API

| Parameter      | Type   | Description                                                                                                                                                               |
|----------------|--------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| error_msg      | String | Error message when calling an API failed. This parameter is unavailable if an API is successfully called.                                                                 |
| error_code     | String | Error code when calling an API failed. For details, see "Error Codes" in <i>ModelArts API Reference</i> . This parameter is unavailable if an API is successfully called. |
| error_solution | String | Solution to an API calling failure. This parameter is unavailable if an API is successfully called.                                                                       |

## 8.1.6 Deleting a Training Job

### Sample Code

In ModelArts notebook, you do not need to enter authentication parameters for session authentication. For details about session authentication of other development environments, see [Session Authentication](#).

- Method 1: Use the specified **job\_id**.

```
from modelarts.session import Session
from modelarts.estimatorV2 import Estimator
session = Session()
Estimator.delete_job_by_id(session=session, job_id="your job id")
```

- Method 2: Use the training job created in [Creating a Training Job](#).

```
job_instance.delete_job()
```

### Parameters

**Table 8-122** delete\_job\_by\_id request parameters

| Parameter | Mandatory | Type   | Description                                                                                                                                                                                                                                         |
|-----------|-----------|--------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| session   | Yes       | Object | Session object. For details about the initialization method, see <a href="#">Session Authentication</a> .                                                                                                                                           |
| job_id    | Yes       | String | ID of a training job. You can obtain <b>job_id</b> using the training job created in <a href="#">Creating a Training Job</a> , for example, <b>job_instance.job_id</b> , or from the response obtained in <a href="#">Obtaining Training Jobs</a> . |

There is no response for successfully calling an API.

**Table 8-123** Response for the failure to call a training API

| Parameter      | Type   | Description                                                                                                                                                               |
|----------------|--------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| error_msg      | String | Error message when calling an API failed. This parameter is unavailable if an API is successfully called.                                                                 |
| error_code     | String | Error code when calling an API failed. For details, see "Error Codes" in <i>ModelArts API Reference</i> . This parameter is unavailable if an API is successfully called. |
| error_solution | String | Solution to an API calling failure. This parameter is unavailable if an API is successfully called.                                                                       |

## 8.1.7 Terminating a Training Job

Terminate a training job. Only jobs in the creating, awaiting, or running state can be terminated.

### Sample Code

In ModelArts notebook, you do not need to enter authentication parameters for session authentication. For details about session authentication of other development environments, see [Session Authentication](#).

- Method 1: Use the specified **job\_id**.  

```
from modelarts.session import Session
from modelarts.estimatorV2 import Estimator
session = Session()
info = Estimator.control_job_by_id(session=session, job_id="your job id")
print(info)
```
- Method 2: Use the training job created in [Creating a Training Job](#).  

```
job_instance.control_job()
```

### Parameters

**Table 8-124** control\_job\_by\_id request parameters

| Parameter | Mandatory | Type   | Description                                                                                                                                                                                                                                         |
|-----------|-----------|--------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| session   | Yes       | Object | Session object. For details about the initialization method, see <a href="#">Session Authentication</a> .                                                                                                                                           |
| job_id    | Yes       | String | ID of a training job. You can obtain <b>job_id</b> using the training job created in <a href="#">Creating a Training Job</a> , for example, <b>job_instance.job_id</b> , or from the response obtained in <a href="#">Obtaining Training Jobs</a> . |

**Table 8-125** Response parameters

| Parameter | Type                                          | Description                                                                                                                                                                                                                                                                                                                                                                                                      |
|-----------|-----------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| kind      | String                                        | Training job type, which defaults to <b>job</b> .<br>Options: <ul style="list-style-type: none"> <li>• <b>job</b>: training job</li> <li>• <b>hetero_job</b>: heterogeneous job</li> <li>• <b>autosearch_job</b>: auto search job</li> <li>• <b>mrs_job</b>: MRS job</li> <li>• <b>edge_job</b>: edge job</li> </ul>                                                                                             |
| metadata  | <a href="#">JobMetadata</a> object            | Metadata of a training job.                                                                                                                                                                                                                                                                                                                                                                                      |
| status    | <a href="#">Status</a> object                 | Status of a training job. When creating a training job, you do not need to set this parameter.                                                                                                                                                                                                                                                                                                                   |
| algorithm | <a href="#">JobAlgorithmResponse</a> object   | Algorithm used by a training job. The following formats are supported: <ul style="list-style-type: none"> <li>• <b>id</b>: Only the algorithm ID is used.</li> <li>• <b>subscription_id</b> and <b>item_version_id</b>: The subscription ID and version ID of the algorithm are used.</li> <li>• <b>code_dir</b> and <b>boot_file</b>: The code directory and boot file of the training job are used.</li> </ul> |
| tasks     | Array of <a href="#">TaskResponse</a> objects | Tasks of a heterogeneous training job.                                                                                                                                                                                                                                                                                                                                                                           |
| spec      | <a href="#">spec</a> object                   | Specifications of a training job.                                                                                                                                                                                                                                                                                                                                                                                |

**Table 8-126** JobMetadata

| Parameter    | Type   | Description                                                                                                                             |
|--------------|--------|-----------------------------------------------------------------------------------------------------------------------------------------|
| id           | String | Training job ID, which is generated and returned by ModelArts after a training job is created.                                          |
| name         | String | Name of a training job. The value must contain 1 to 64 characters consisting of only digits, letters, underscores (_), and hyphens (-). |
| workspace_id | String | Workspace where a training job is deployed. Default value: <b>0</b>                                                                     |
| description  | String | Description of a training job, which defaults to <b>NULL</b> . The value must contain 0 to 256 characters.                              |

| Parameter   | Type               | Description                                                                                                                                                                                |
|-------------|--------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| create_time | Long               | Time when a training job was created, in milliseconds. The value is generated and returned by ModelArts after a training job is created.                                                   |
| user_name   | String             | Username for creating a training job. The username is generated and returned by ModelArts after a training job is created.                                                                 |
| annotations | Map<String,String> | Declaration template of a training job. For heterogeneous jobs, the default value of <b>job_template</b> is <b>Template RL</b> . For other jobs, the default value is <b>Template DL</b> . |

**Table 8-127** Status

| Parameter          | Type                  | Description                                                                                                                                                                                           |
|--------------------|-----------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| phase              | String                | Level-1 status of a training job. The value will remain unchanged. Options: <b>Creating, Pending, Running, Failed, Completed, Terminating, Terminated, and Abnormal</b>                               |
| secondary_phase    | String                | Level-2 status of a training job. The value can be changed. Options: <b>Creating, Queuing, Running, Failed, Completed, Terminating, Terminated, CreateFailed, TerminatedFailed, Unknown, and Lost</b> |
| duration           | Long                  | Running duration of a training job, in milliseconds                                                                                                                                                   |
| node_count_metrics | Array<Array<Integer>> | Node count changes during the runtime of a training job                                                                                                                                               |
| tasks              | Array of strings      | Task of a training job                                                                                                                                                                                |
| start_time         | String                | Start time of a training job. The value is in timestamp format.                                                                                                                                       |
| task_statuses      | Array of objects      | Status of a training job task                                                                                                                                                                         |

**Table 8-128** task\_statuses

| Parameter | Type    | Description                      |
|-----------|---------|----------------------------------|
| task      | String  | Task of a training job           |
| exit_code | Integer | Exit code of a training job task |

| Parameter | Type   | Description                          |
|-----------|--------|--------------------------------------|
| message   | String | Error message of a training job task |

**Table 8-129** JobAlgorithmResponse

| Parameter                 | Type   | Description                                                                                                                                                                                                                                                                                                                                                        |
|---------------------------|--------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| id                        | String | Algorithm ID<br>Options: <ul style="list-style-type: none"> <li>• <b>id</b>: Only the algorithm ID is used.</li> <li>• <b>subscription_id</b> and <b>item_version_id</b>: The subscription ID and version ID of the algorithm are used.</li> <li>• <b>code_dir</b> and <b>boot_file</b>: The code directory and boot file of the training job are used.</li> </ul> |
| name                      | String | Algorithm name                                                                                                                                                                                                                                                                                                                                                     |
| subscription_id           | String | Subscription ID of the subscribed algorithm, which must be used with <b>item_version_id</b>                                                                                                                                                                                                                                                                        |
| item_version_id           | String | Version ID of the subscribed algorithm, which must be used with <b>subscription_id</b>                                                                                                                                                                                                                                                                             |
| code_dir                  | String | Code directory of a training job, for example, <code>/usr/app/</code> . This parameter must be used with <b>boot_file</b> . Leave this parameter blank if <b>id</b> , or <b>subscription_id</b> and <b>item_version_id</b> are specified.                                                                                                                          |
| boot_file                 | String | Boot file of a training job, which must be stored in the code directory, for example, <code>/usr/app/boot.py</code> . This parameter must be used with <b>code_dir</b> . Leave this parameter blank if <b>id</b> , or <b>subscription_id</b> and <b>item_version_id</b> are specified.                                                                             |
| autosearch_config_path    | String | YAML configuration path of an auto search job. An OBS URL is required.                                                                                                                                                                                                                                                                                             |
| autosearch_framework_path | String | Framework code directory of an auto search job. An OBS URL is required.                                                                                                                                                                                                                                                                                            |
| command                   | String | Boot command for starting the container of the custom image used for creating a training job. The value of this parameter can be the same as the <b>code_dir</b> value.                                                                                                                                                                                            |



| Parameter    | Type                              | Description                                                                                                                                                                                                            |
|--------------|-----------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| parameters   | Array of <b>Parameter</b> objects | Running parameters of a training job.                                                                                                                                                                                  |
| policies     | <b>policies</b> object            | Policies supported by a training job.                                                                                                                                                                                  |
| inputs       | Array of <b>Input</b> objects     | Input of a training job.                                                                                                                                                                                               |
| outputs      | Array of <b>Output</b> objects    | Output of a training job.                                                                                                                                                                                              |
| engine       | <b>engine</b> object              | Engine of a training job. Leave this parameter blank if the job is created using <b>id</b> of the algorithm in algorithm management, or <b>subscription_id</b> and <b>item_version_id</b> of the subscribed algorithm. |
| environments | Map<String,String>                | Environment variables of a training job in the format of "key":"value". Leave this parameter blank.                                                                                                                    |

**Table 8-130** Parameter

| Parameter        | Type                           | Description                      |
|------------------|--------------------------------|----------------------------------|
| name             | String                         | Parameter name                   |
| value            | String                         | Parameter value                  |
| description      | String                         | Parameter description            |
| constraint       | <b>constraint</b> object       | Parameter constraint             |
| i18n_description | <b>i18n_description</b> object | Internationalization description |

**Table 8-131** constraint

| Parameter | Type    | Description                        |
|-----------|---------|------------------------------------|
| type      | String  | Parameter type                     |
| editable  | Boolean | Whether the parameter is editable  |
| required  | Boolean | Whether the parameter is mandatory |
| sensitive | Boolean | Whether the parameter is sensitive |

| Parameter   | Type             | Description |
|-------------|------------------|-------------|
| valid_type  | String           | Valid type  |
| valid_range | Array of strings | Valid range |

**Table 8-132** i18n\_description

| Parameter   | Type   | Description                   |
|-------------|--------|-------------------------------|
| language    | String | Internationalization language |
| description | String | Description                   |

**Table 8-133** policies

| Parameter   | Type                               | Description                         |
|-------------|------------------------------------|-------------------------------------|
| auto_search | <a href="#">auto_search</a> object | Hyperparameter search configuration |

**Table 8-134** auto\_search

| Parameter          | Type             | Description                                       |
|--------------------|------------------|---------------------------------------------------|
| skip_search_params | String           | Hyperparameter parameters that need to be skipped |
| reward_attrs       | Array of objects | Search metrics                                    |
| search_params      | Array of objects | Search parameters                                 |
| algo_configs       | Array of objects | Search algorithm configurations                   |

**Table 8-135** reward\_attrs

| Parameter | Type   | Description                                                                                                                                                                 |
|-----------|--------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| name      | String | Metric name                                                                                                                                                                 |
| mode      | String | Search mode <ul style="list-style-type: none"> <li>• <b>max</b>: A larger metric value is preferred.</li> <li>• <b>min</b>: A smaller metric value is preferred.</li> </ul> |

| Parameter | Type   | Description                    |
|-----------|--------|--------------------------------|
| regex     | String | Regular expression of a metric |

**Table 8-136** search\_params

| Parameter           | Type             | Description                                                                                                                                                                       |
|---------------------|------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| name                | String           | Hyperparameter name                                                                                                                                                               |
| param_type          | String           | Parameter type <ul style="list-style-type: none"> <li>• <b>continuous</b>: Parameter values are continuous.</li> <li>• <b>discrete</b>: Parameter values are discrete.</li> </ul> |
| lower_bound         | String           | Lower bound of the hyperparameter                                                                                                                                                 |
| upper_bound         | String           | Upper bound of the hyperparameter                                                                                                                                                 |
| discrete_points_num | String           | Number of discrete points of a hyperparameter with continuous values                                                                                                              |
| discrete_values     | Array of strings | Discrete hyperparameter values                                                                                                                                                    |

**Table 8-137** algo\_configs

| Parameter | Type                                                           | Description                  |
|-----------|----------------------------------------------------------------|------------------------------|
| name      | String                                                         | Name of the search algorithm |
| params    | Array of <a href="#">AutoSearchAlgoConfigParameter</a> objects | Search algorithm parameters  |

**Table 8-138** AutoSearchAlgoConfigParameter

| Parameter | Type   | Description     |
|-----------|--------|-----------------|
| key       | String | Parameter key   |
| value     | String | Parameter value |
| type      | String | Parameter type  |

**Table 8-139** Input

| Parameter         | Type                        | Description                                                                |
|-------------------|-----------------------------|----------------------------------------------------------------------------|
| name              | String                      | Name of the data input channel                                             |
| description       | String                      | Description of the data input channel                                      |
| local_dir         | String                      | Local directory of the container to which the data input channel is mapped |
| remote            | <b>InputDataInfo</b> object | Information of the data input                                              |
| remote_constraint | Array of objects            | Data input constraint                                                      |

**Table 8-140** InputDataInfo

| Parameter | Type                  | Description                                   |
|-----------|-----------------------|-----------------------------------------------|
| dataset   | <b>dataset</b> object | Dataset as the data input                     |
| obs       | <b>obs</b> object     | OBS in which data input and output are stored |

**Table 8-141** dataset

| Parameter  | Type   | Description                                                                                                                                                           |
|------------|--------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| id         | String | Dataset ID of a training job                                                                                                                                          |
| version_id | String | Dataset version ID of a training job                                                                                                                                  |
| obs_url    | String | OBS URL of the dataset for a training job, which is automatically parsed by ModelArts based on the dataset ID and dataset version IDs, for example, <b>/usr/data/</b> |

**Table 8-142** obs

| Parameter | Type   | Description                                                               |
|-----------|--------|---------------------------------------------------------------------------|
| obs_url   | String | OBS URL of the dataset for a training job, for example, <b>/usr/data/</b> |

**Table 8-143** remote\_constraint

| Parameter  | Type   | Description                                                                                                                                                                                                                                                |
|------------|--------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| data_type  | String | Data input type, including the data storage location and dataset                                                                                                                                                                                           |
| attributes | String | Attributes when a dataset functions as the data input<br>Options: <ul style="list-style-type: none"> <li>• <b>data_format</b>: data format</li> <li>• <b>data_segmentation</b>: data segmentation</li> <li>• <b>dataset_type</b>: data labeling</li> </ul> |

**Table 8-144** Output

| Parameter   | Type                 | Description                                                                 |
|-------------|----------------------|-----------------------------------------------------------------------------|
| name        | String               | Name of the data output channel                                             |
| description | String               | Description of the data output channel                                      |
| local_dir   | String               | Local directory of the container to which the data output channel is mapped |
| remote      | <b>remote</b> object | Information of the data output                                              |

**Table 8-145** remote

| Parameter | Type              | Description                   |
|-----------|-------------------|-------------------------------|
| obs       | <b>obs</b> object | OBS to which data is exported |

**Table 8-146** obs

| Parameter | Type   | Description                       |
|-----------|--------|-----------------------------------|
| obs_url   | String | OBS URL to which data is exported |

**Table 8-147** engine

| Parameter      | Type   | Description                                                                                                                               |
|----------------|--------|-------------------------------------------------------------------------------------------------------------------------------------------|
| engine_id      | String | Engine ID selected for a training job, which can be <b>engine_id</b> , <b>engine_name</b> and <b>engine_version</b> , or <b>image_url</b> |
| engine_name    | String | Name of the engine selected for a training job. Leave this parameter blank if <b>engine_id</b> is specified.                              |
| engine_version | String | Version of the engine selected for a training job. Leave this parameter blank if <b>engine_id</b> is specified.                           |
| image_url      | String | Custom image URL selected for a training job                                                                                              |

**Table 8-148** TaskResponse

| Parameter     | Type                         | Description                                                                                                                                                                                            |
|---------------|------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| role          | String                       | Role of a heterogeneous training job task<br>Options:<br><ul style="list-style-type: none"> <li>• <b>learner</b>: GPUs or CPUs are supported.</li> <li>• <b>worker</b>: CPUs are supported.</li> </ul> |
| algorithm     | <b>algorithm</b> object      | Algorithm configurations in algorithm management                                                                                                                                                       |
| task_resource | <b>FlavorResponse</b> object | Flavors for a training job or an algorithm                                                                                                                                                             |

**Table 8-149** algorithm

| Parameter | Type                  | Description                                                            |
|-----------|-----------------------|------------------------------------------------------------------------|
| code_dir  | String                | Absolute path of the directory where the algorithm boot file is stored |
| boot_file | String                | Absolute path of the algorithm boot file                               |
| inputs    | <b>inputs</b> object  | Algorithm input channel                                                |
| outputs   | <b>outputs</b> object | Algorithm output channel                                               |
| engine    | <b>engine</b> object  | Engine on which a heterogeneous job depends                            |

**Table 8-150** inputs

| Parameter | Type                 | Description                                                                        |
|-----------|----------------------|------------------------------------------------------------------------------------|
| name      | String               | Name of the data input channel                                                     |
| local_dir | String               | Local path of the container to which the data input and output channels are mapped |
| remote    | <b>remote</b> object | Actual data input, which can only be OBS for heterogeneous jobs                    |

**Table 8-151** remote

| Parameter | Type              | Description                                   |
|-----------|-------------------|-----------------------------------------------|
| obs       | <b>obs</b> object | OBS in which data input and output are stored |

**Table 8-152** obs

| Parameter | Type   | Description                                                               |
|-----------|--------|---------------------------------------------------------------------------|
| obs_url   | String | OBS URL of the dataset for a training job, for example, <b>/usr/data/</b> |

**Table 8-153** outputs

| Parameter | Type                 | Description                                                                 |
|-----------|----------------------|-----------------------------------------------------------------------------|
| name      | String               | Name of the data output channel                                             |
| local_dir | String               | Local directory of the container to which the data output channel is mapped |
| remote    | <b>remote</b> object | Information of the data output                                              |
| mode      | String               | Data transmission mode, which defaults to <b>upload_periodically</b>        |
| period    | String               | Data transmission period, which defaults to <b>30s</b>                      |

**Table 8-154** remote

| Parameter | Type                       | Description                   |
|-----------|----------------------------|-------------------------------|
| obs       | <a href="#">obs</a> object | OBS to which data is exported |

**Table 8-155** obs

| Parameter | Type   | Description                       |
|-----------|--------|-----------------------------------|
| obs_url   | String | OBS URL to which data is exported |

**Table 8-156** engine

| Parameter      | Type    | Description                                                                 |
|----------------|---------|-----------------------------------------------------------------------------|
| engine_id      | String  | Engine ID of a heterogeneous job, for example, <b>caffe-1.0.0-python2.7</b> |
| engine_name    | String  | Engine name of a heterogeneous job, for example, <b>Caffe</b>               |
| engine_version | String  | Engine version of a heterogeneous job                                       |
| v1_compatible  | Boolean | Whether v1 is compatible                                                    |
| run_user       | String  | User UID for which the engine is started by default                         |

**Table 8-157** FlavorResponse

| Parameter   | Type    | Description                                                                                                             |
|-------------|---------|-------------------------------------------------------------------------------------------------------------------------|
| flavor_id   | String  | ID of the resource flavor                                                                                               |
| flavor_name | String  | Name of the resource flavor                                                                                             |
| max_num     | Integer | Maximum number of nodes with the resource flavor                                                                        |
| flavor_type | String  | Resource flavor type. Options: <ul style="list-style-type: none"> <li>• CPU</li> <li>• GPU</li> <li>• Ascend</li> </ul> |



| Parameter   | Type                      | Description                              |
|-------------|---------------------------|------------------------------------------|
| billing     | <b>billing</b> object     | Billing information of a resource flavor |
| flavor_info | <b>flavor_info</b> object | Resource flavor details                  |
| attributes  | Map<String,String>        | Other flavor attributes                  |

**Table 8-158** billing

| Parameter | Type    | Description             |
|-----------|---------|-------------------------|
| code      | String  | Billing code            |
| unit_num  | Integer | Number of billing units |

**Table 8-159** flavor\_info

| Parameter | Type                 | Description                                                                                                        |
|-----------|----------------------|--------------------------------------------------------------------------------------------------------------------|
| max_num   | Integer              | Maximum number of nodes that can be selected. Value <b>1</b> indicates that the distributed mode is not supported. |
| cpu       | <b>cpu</b> object    | CPU specifications                                                                                                 |
| gpu       | <b>gpu</b> object    | GPU specifications                                                                                                 |
| npu       | <b>npu</b> object    | Ascend specifications                                                                                              |
| memory    | <b>memory</b> object | Memory information                                                                                                 |

**Table 8-160** cpu

| Parameter | Type    | Description      |
|-----------|---------|------------------|
| arch      | String  | CPU architecture |
| core_num  | Integer | Number of cores  |

**Table 8-161** gpu

| Parameter    | Type    | Description    |
|--------------|---------|----------------|
| unit_num     | Integer | Number of GPUs |
| product_name | String  | Product name   |
| memory       | String  | Memory         |

**Table 8-162** npu

| Parameter    | Type   | Description    |
|--------------|--------|----------------|
| unit_num     | String | Number of NPUs |
| product_name | String | Product name   |
| memory       | String | Memory         |

**Table 8-163** memory

| Parameter | Type    | Description            |
|-----------|---------|------------------------|
| size      | Integer | Memory size            |
| unit      | String  | Number of memory units |

**Table 8-164** spec

| Parameter       | Type                                   | Description                                                                                                     |
|-----------------|----------------------------------------|-----------------------------------------------------------------------------------------------------------------|
| resource        | <a href="#">Resource</a> object        | Resource flavors of a training job, which can either be <b>flavor_id</b> or <b>pool_id</b> and <b>flavor_id</b> |
| volumes         | Array of objects                       | Volumes attached for a training job                                                                             |
| log_export_path | <a href="#">log_export_path</a> object | Export path of training job logs                                                                                |

**Table 8-165** Resource

| Parameter | Type   | Description                                                                                          |
|-----------|--------|------------------------------------------------------------------------------------------------------|
| policy    | String | Resource flavor mode of a training job. Options: <b>regular</b> , <b>economic</b> , and <b>turbo</b> |

| Parameter     | Type                        | Description                                                                    |
|---------------|-----------------------------|--------------------------------------------------------------------------------|
| flavor_id     | String                      | Resource flavor ID of a training job                                           |
| flavor_name   | String                      | Read-only flavor name returned by ModelArts when <b>flavor_id</b> is specified |
| node_count    | Integer                     | Number of resource replicas selected for a training job<br>Minimum value: 1    |
| pool_id       | String                      | Resource pool ID selected for a training job                                   |
| flavor_detail | <b>flavor_detail</b> object | Flavors for a training job or an algorithm                                     |

**Table 8-166** flavor\_detail

| Parameter   | Type                      | Description                                                                                                             |
|-------------|---------------------------|-------------------------------------------------------------------------------------------------------------------------|
| flavor_type | String                    | Resource flavor type. Options: <ul style="list-style-type: none"> <li>• CPU</li> <li>• GPU</li> <li>• Ascend</li> </ul> |
| billing     | <b>billing</b> object     | Billing information of a resource flavor                                                                                |
| flavor_info | <b>flavor_info</b> object | Resource flavor details                                                                                                 |

**Table 8-167** billing

| Parameter | Type    | Description             |
|-----------|---------|-------------------------|
| code      | String  | Billing code            |
| unit_num  | Integer | Number of billing units |

**Table 8-168** flavor\_info

| Parameter | Type              | Description                                                                                                 |
|-----------|-------------------|-------------------------------------------------------------------------------------------------------------|
| max_num   | Integer           | Maximum number of nodes that can be selected. Value 1 indicates that the distributed mode is not supported. |
| cpu       | <b>cpu</b> object | CPU specifications                                                                                          |

| Parameter | Type                 | Description           |
|-----------|----------------------|-----------------------|
| gpu       | <b>gpu</b> object    | GPU specifications    |
| npu       | <b>npu</b> object    | Ascend specifications |
| memory    | <b>memory</b> object | Memory information    |
| disk      | <b>disk</b> object   | Disk information      |

**Table 8-169** cpu

| Parameter | Type    | Description      |
|-----------|---------|------------------|
| arch      | String  | CPU architecture |
| core_num  | Integer | Number of cores  |

**Table 8-170** gpu

| Parameter    | Type    | Description    |
|--------------|---------|----------------|
| unit_num     | Integer | Number of GPUs |
| product_name | String  | Product name   |
| memory       | String  | Memory         |

**Table 8-171** npu

| Parameter    | Type   | Description    |
|--------------|--------|----------------|
| unit_num     | String | Number of NPUs |
| product_name | String | Product name   |
| memory       | String | Memory         |

**Table 8-172** memory

| Parameter | Type    | Description |
|-----------|---------|-------------|
| size      | Integer | Memory size |

| Parameter | Type   | Description            |
|-----------|--------|------------------------|
| unit      | String | Number of memory units |

**Table 8-173** disk

| Parameter | Type   | Description                                  |
|-----------|--------|----------------------------------------------|
| size      | String | Disk size                                    |
| unit      | String | Unit of the disk size, which is GB generally |

**Table 8-174** volumes

| Parameter | Type                          | Description                |
|-----------|-------------------------------|----------------------------|
| nfs       | <a href="#">nfs</a><br>object | Disks attached in NFS mode |

**Table 8-175** nfs

| Parameter       | Type    | Description                                                           |
|-----------------|---------|-----------------------------------------------------------------------|
| nfs_server_path | String  | NFS server path                                                       |
| local_path      | String  | Path for attaching disks to the training container                    |
| read_only       | Boolean | Whether the disks attached to the container in NFS mode are read-only |

**Table 8-176** log\_export\_path

| Parameter | Type   | Description                                         |
|-----------|--------|-----------------------------------------------------|
| obs_url   | String | OBS URL for storing training job logs               |
| host_path | String | Path of the host where training job logs are stored |

**Table 8-177** Response for the failure to call a training API

| Parameter | Type   | Description                                                                                               |
|-----------|--------|-----------------------------------------------------------------------------------------------------------|
| error_msg | String | Error message when calling an API failed. This parameter is unavailable if an API is successfully called. |

| Parameter      | Type   | Description                                                                                                                                                               |
|----------------|--------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| error_code     | String | Error code when calling an API failed. For details, see "Error Codes" in <i>ModelArts API Reference</i> . This parameter is unavailable if an API is successfully called. |
| error_solution | String | Solution to an API calling failure. This parameter is unavailable if an API is successfully called.                                                                       |

## 8.1.8 Obtaining Training Logs

### Sample Code

In ModelArts notebook, you do not need to enter authentication parameters for session authentication. For details about session authentication of other development environments, see [Session Authentication](#).

- Method 1: Use the specified **job\_id**.  

```
from modelarts.session import Session
from modelarts.estimatorV2 import Estimator
session = Session()
estimator = Estimator(session=session, job_id="your job id")
info = estimator.get_job_log()
print(info)
```
- Method 2: Use the training job created in [Creating a Training Job](#).  

```
log = job_instance.get_job_log(task_id="worker-0")
print(log)
```

### Parameters

**Table 8-178** Parameters for initializing the Estimator

| Parameter | Mandatory | Type   | Description                                                                                                                                                                                                                                         |
|-----------|-----------|--------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| session   | Yes       | Object | Session object. For details about the initialization method, see <a href="#">Session Authentication</a> .                                                                                                                                           |
| job_id    | Yes       | String | ID of a training job. You can obtain <b>job_id</b> using the training job created in <a href="#">Creating a Training Job</a> , for example, <b>job_instance.job_id</b> , or from the response obtained in <a href="#">Obtaining Training Jobs</a> . |

**Table 8-179** get\_job\_log request parameters

| Parameter | Mandatory | Type   | Description                                                                                                                                                                                                                        |
|-----------|-----------|--------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| task_id   | No        | String | ID of a worker node for obtaining logs. It defaults to <b>worker-0</b> . If <b>train_instance_count</b> is set to <b>2</b> when you create a training job, the value of this parameter can be <b>worker-0</b> or <b>worker-1</b> . |

**Table 8-180** Response parameters

| Parameter    | Type    | Description                                                                                                                                                                                                                                                          |
|--------------|---------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| content      | String  | Log content <ul style="list-style-type: none"> <li>If the size of the log file does not exceed the limit allowed (n MB), all logs are returned.</li> <li>If the size of the log file exceeds the limit allowed (n MB), the latest n MB logs are returned.</li> </ul> |
| current_size | Integer | Size of the returned log file, in bytes. The maximum value is 5 MB.                                                                                                                                                                                                  |
| full_size    | Integer | Size of a complete log file, in bytes.                                                                                                                                                                                                                               |

**Table 8-181** Response for the failure to call a training API

| Parameter      | Type   | Description                                                                                                                                                               |
|----------------|--------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| error_message  | String | Error message when calling an API failed. This parameter is unavailable if an API is successfully called.                                                                 |
| error_code     | String | Error code when calling an API failed. For details, see "Error Codes" in <i>ModelArts API Reference</i> . This parameter is unavailable if an API is successfully called. |
| error_solution | String | Solution to an API calling failure. This parameter is unavailable if an API is successfully called.                                                                       |

## 8.1.9 Obtaining the Runtime Metrics of a Training Job

### Sample Code

In ModelArts notebook, you do not need to enter authentication parameters for session authentication. For details about session authentication of other development environments, see [Session Authentication](#).

- Method 1: Use the specified `job_id`.

```
from modelarts.session import Session
from modelarts.estimatorV2 import Estimator
session = Session()
estimator = Estimator(session=session, job_id="your job id")
info = estimator.get_job_metrics()
print(info)
```

- Method 2: Use the training job created in [Creating a Training Job](#).

```
info = job_instance.get_job_metrics(task_id="worker-0")
print(info)
```

### Parameters

**Table 8-182** Parameters for initializing the Estimator

| Parameter | Mandatory | Type   | Description                                                                                                                                                                                                                                                     |
|-----------|-----------|--------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| session   | Yes       | Object | Session object. For details about the initialization method, see <a href="#">Session Authentication</a> .                                                                                                                                                       |
| job_id    | Yes       | String | ID of a training job. You can obtain <code>job_id</code> using the training job created in <a href="#">Creating a Training Job</a> , for example, <code>job_instance.job_id</code> , or from the response obtained in <a href="#">Obtaining Training Jobs</a> . |

**Table 8-183** `get_job_log` request parameters

| Parameter | Mandatory | Type   | Description                                                                                                                                                                                                                                         |
|-----------|-----------|--------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| task_id   | No        | String | ID of a worker node for obtaining logs. It defaults to <code>worker-0</code> . If <code>train_instance_count</code> is set to 2 when you create a training job, the value of this parameter can be <code>worker-0</code> or <code>worker-1</code> . |



**Table 8-184** Response parameters

| Parameter | Type             | Description     |
|-----------|------------------|-----------------|
| metrics   | Array of objects | Runtime metrics |

**Table 8-185** metrics

| Parameter | Type             | Description                                                                                                                                                                                           |
|-----------|------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| metric    | String           | Runtime metric. The value can be cpuUsage (CPU usage), memUsage (physical memory usage), gpuUtil (GPU usage), gpuMemUsage (GPU memory usage), npuUtil (NPU usage), or npuMemUsage (NPU memory usage). |
| value     | Array of numbers | Value of a runtime metric. An average value is collected every minute.                                                                                                                                |

**Table 8-186** Response for the failure to call a training API

| Parameter      | Type   | Description                                                                                                                                                               |
|----------------|--------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| error_msg      | String | Error message when calling an API failed. This parameter is unavailable if an API is successfully called.                                                                 |
| error_code     | String | Error code when calling an API failed. For details, see "Error Codes" in <i>ModelArts API Reference</i> . This parameter is unavailable if an API is successfully called. |
| error_solution | String | Solution to an API calling failure. This parameter is unavailable if an API is successfully called.                                                                       |

## 8.2 APIs for Resources and Engine Specifications

### 8.2.1 Obtaining Resource Flavors

#### Sample Code

In ModelArts notebook, you do not need to enter authentication parameters for session authentication. For details about session authentication of other development environments, see [Session Authentication](#).

```
from modelarts.session import Session
from modelarts.estimatorV2 import Estimator
session = Session()
info = Estimator.get_train_instance_types(session=session)
print(info)
```

## Parameters

**Table 8-187** get\_train\_instance\_types parameters

| Parameter | Mandatory | Type   | Description                                                                                               |
|-----------|-----------|--------|-----------------------------------------------------------------------------------------------------------|
| session   | Yes       | Object | Session object. For details about the initialization method, see <a href="#">Session Authentication</a> . |

**Table 8-188** Successful response parameters

| Type | Description              |
|------|--------------------------|
| List | List of resource flavors |

**Table 8-189** Response for the failure to call a training API

| Parameter      | Type   | Description                                                                                                                                                               |
|----------------|--------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| error_msg      | String | Error message when calling an API failed. This parameter is unavailable if an API is successfully called.                                                                 |
| error_code     | String | Error code when calling an API failed. For details, see "Error Codes" in <i>ModelArts API Reference</i> . This parameter is unavailable if an API is successfully called. |
| error_solution | String | Solution to an API calling failure. This parameter is unavailable if an API is successfully called.                                                                       |

## 8.2.2 Obtaining Engine Types

### Sample Code

In ModelArts notebook, you do not need to enter authentication parameters for session authentication. For details about session authentication of other development environments, see [Session Authentication](#).

```
from modelarts.session import Session
from modelarts.estimatorV2 import Estimator
session = Session()
info = Estimator.get_framework_list(session=session)
print(info)
```

## Parameters

**Table 8-190** `get_train_instance_types` parameters

| Parameter | Mandatory | Type   | Description                                                                                               |
|-----------|-----------|--------|-----------------------------------------------------------------------------------------------------------|
| session   | Yes       | Object | Session object. For details about the initialization method, see <a href="#">Session Authentication</a> . |

**Table 8-191** Successful response parameters of `get_framework_list`

| Type | Description                                                      |
|------|------------------------------------------------------------------|
| List | List of engine types. For details, see <a href="#">Table 3</a> . |

**Table 8-192** `framework_list` parameters

| Parameter         | Type   | Description    |
|-------------------|--------|----------------|
| framework_type    | String | Engine type    |
| framework_version | String | Engine version |

**Table 8-193** Response for the failure to call a training API

| Parameter      | Type   | Description                                                                                                                                                               |
|----------------|--------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| error_msg      | String | Error message when calling an API failed. This parameter is unavailable if an API is successfully called.                                                                 |
| error_code     | String | Error code when calling an API failed. For details, see "Error Codes" in <i>ModelArts API Reference</i> . This parameter is unavailable if an API is successfully called. |
| error_solution | String | Solution to an API calling failure. This parameter is unavailable if an API is successfully called.                                                                       |

# 9 Model Management

## 9.1 Importing a Model

Importing a model includes:

- Initialize the existing model and create a model object based on the model ID.
- Create a model. For details about the attributes of the created model, see [Obtaining Details About a Model](#).

### Sample Model File

The following uses PyTorch as an example to describe how to edit a model file. For details about the PyTorch model package structure, see "Model Inference > Inference Specifications > Model Package Specifications > Introduction to Model Package Specifications" in *ModelArts User Guide*.

For details about PyTorch custom scripts, see "Model Inference > Inference Specifications > PyTorch" in *ModelArts User Guide*.

```
OBS bucket or directory name
├── resnet
│ ├── model Mandatory: Fixed subdirectory name. The subdirectory is used to store model-related files.
│ │ └── <<Custom Python package>> (Optional) Custom Python package, which can be directly
│ │ referenced in model inference code
│ │ ├── mnist_mlp.pt (Mandatory) PyTorch model file, which contains variable and weight information
│ │ │ and is saved as state_dict
│ │ ├── config.json Mandatory: Model configuration file. The file name is fixed to config.json. Only one
│ │ │ model configuration file is allowed.
│ │ └── customize_service.py Mandatory: Model inference code. The file name is fixed to
│ │ customize_service.py. Only one model inference file is allowed. The files on which customize_service.py
│ │ depends can be directly stored in the model directory.
```

### Sample Code

In ModelArts notebook, you do not need to enter authentication parameters for session authentication. For details about session authentication of other development environments, see [Session Authentication](#).

```
from modelarts.session import Session
from modelarts.model import Model
from modelarts.config.model_config import ServiceConfig, Params, Dependencies, Packages
```

```
session = Session()
```

- Method 1: Initialize an existing model.

```
model_instance = Model(session, model_id="your_model_id")
```

- Method 2: Create a model.

- Use a preset image and specify an OBS path to create a model.

```
model_location = "/your_obs_bucket/model_path" # Change to the OBS path to the
model file
execution_code = "/your_obs_bucket/model_path/customize_service.py"
runtime = "python3.7"
```

```
model_instance = Model(
 session,
 model_name="input_model_name", # (Optional) Model name
 model_version="1.0.0", # (Optional) Model version
 source_location=model_location, # OBS path to the model file, for example, /
your_obs_bucket/model_path
 model_type="PyTorch", # Model type
 execution_code=execution_code, # (Optional) OBS path to the execution
script, for example, /your_obs_bucket/model_path/customize_service.py
 runtime = runtime # (Optional) Supported runtime environment
)
```

#### NOTE

**dependencies** will overwrite the data in **config.json** in the preceding example. You do not need to use **dependencies**. The following section describes the **dependencies** formats.

- Format of the **dependencies** parameter group

SDKs define the **dependencies** parameter group. **dependencies** is in list format, and those of the tuple objects in the list are Dependencies.

The code is as follows:

```
dependencies = []
dependency1 = Dependencies(
 installer="pip", # Installation mode. pip is supported.
 packages=packages # Collection of dependency packages. For details, see
packages.
)
dependencies.append(dependency1)
```

- Format of the **package** parameter group

SDKs define the **packages** parameter group. **packages** is in list format, and those of the tuple objects in the list are Packages.

The code is as follows:

```
packages = []
package1 = Packages(
 package_name="package_name", # Package name
 package_version="version", # Package version
 restraint="EXACT"
)
packages.append(package1)
```

 NOTE

The following is an example of creating a **dependencies** parameter group:

```
dependencies = []
packages = [{
 "package_name": "numpy",
 "package_version": "1.15.0",
 "restraint": "EXACT"
}, {
 "package_name": "h5py",
 "package_version": "2.8.0",
 "restraint": "EXACT"
}]
dependency = Dependencies(installer="pip", packages=packages)
dependencies.append(dependency)
```

- Use a custom image to create a model.

This method applies if the script of the inference service has been built in the custom image and the service is automatically started when the image is started.

```
from modelarts.session import Session
from modelarts.model import Model

session = Session()
image_path = "custom_image_path"
model_instance = Model(
 session,
 model_name="your_model_name", # Model name
 model_version="0.1.0", # Model version
 source_location=image_path, # Model file path
 model_type="Image" # Model type
)
```

## Parameters

**Table 9-1** Parameters for initializing a model

| Parameter | Mandatory | Type   | Description                                                                                               |
|-----------|-----------|--------|-----------------------------------------------------------------------------------------------------------|
| session   | Yes       | Object | Session object. For details about the initialization method, see <a href="#">Session Authentication</a> . |
| model_id  | Yes       | String | Model ID                                                                                                  |

**Table 9-2** Parameters for creating a model

| Parameter | Mandatory | Type   | Description                                                                                               |
|-----------|-----------|--------|-----------------------------------------------------------------------------------------------------------|
| session   | Yes       | Object | Session object. For details about the initialization method, see <a href="#">Session Authentication</a> . |

| Parameter            | Mandatory | Type                 | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
|----------------------|-----------|----------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| model_name           | No        | String               | Name of a model that consists of 1 to 64 characters. If this parameter is not specified, the system automatically generates a model name.                                                                                                                                                                                                                                                                                                                      |
| model_version        | Yes       | String               | Model version in the format of <i>Digit.Digit.Digit</i> . The value range of the digits is [1, 99]. The version number cannot start with 0, for example, <b>01.01.01</b> .                                                                                                                                                                                                                                                                                     |
| publish              | No        | Bool                 | Whether to publish a model. The options are as follows: <ul style="list-style-type: none"> <li>• <b>True</b>: Publish the model. (Default value)</li> <li>• <b>False</b>: Do not publish the model.</li> </ul>                                                                                                                                                                                                                                                 |
| source_location_type | No        | String               | Model location type. The options are as follows: <ul style="list-style-type: none"> <li>• <b>OBS_SOURCE</b>: OBS path. (Default value)</li> <li>• <b>LOCAL_SOURCE</b>: local path.</li> </ul>                                                                                                                                                                                                                                                                  |
| source_location      | Yes       | String               | Path (parent directory) of the model file <ul style="list-style-type: none"> <li>• If <b>source_location_type</b> is set to <b>OBS_SOURCE</b>, the model file path is an OBS path in the format of / <b>obs_bucketname</b>/.../ <b>model_file_parent_dir</b>/.</li> <li>• If <b>source_location_type</b> is set to <b>LOCAL_SOURCE</b>, the model file path is a local path in the format of / <b>local_path</b>/.../<b>model_file_parent_dir</b>/.</li> </ul> |
| environment          | No        | Environment instance | Environment required for normal model running, such as the Python or TensorFlow version                                                                                                                                                                                                                                                                                                                                                                        |
| source_job_id        | No        | String               | ID of the source training job. If the model is generated from a training job, specify this parameter for source tracing. If the model is imported from a third-party meta model, leave this parameter blank. By default, this parameter is left blank.                                                                                                                                                                                                         |
| source_job_version   | No        | String               | Version of the source training job. If the model is generated from a training job, specify this parameter for source tracing. If the model is imported from a third-party meta model, leave this parameter blank. By default, this parameter is left blank.                                                                                                                                                                                                    |

| Parameter       | Mandatory | Type                | Description                                                                                                                                                                                                                                                                                                                                                                |
|-----------------|-----------|---------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| source_type     | No        | String              | Model source type. If the model is deployed via a training job, leave this parameter blank. By default, this parameter is left blank.                                                                                                                                                                                                                                      |
| model_type      | Yes       | String              | Model type. The value can be <b>TensorFlow</b> , <b>MXNet</b> , <b>Spark_MLlib</b> , <b>Scikit_Learn</b> , <b>XGBoost</b> , <b>MindSpore</b> , <b>Image</b> , or <b>PyTorch</b> .                                                                                                                                                                                          |
| model_algorithm | No        | String              | Model algorithm. If the algorithm has been configured in the model configuration file, this parameter can be left blank. For example, <b>predict_analysis</b> , <b>object_detection</b> , or <b>image_classification</b> .                                                                                                                                                 |
| description     | No        | String              | Model description, which contains a maximum of 100 characters and cannot contain the following special characters: ! <>=&"                                                                                                                                                                                                                                                 |
| execution_code  | No        | String              | OBS path to the script to be executed. If <b>customize_service.py</b> is not output by the model, configure this parameter to specify the path. The inference script must be stored in the <b>model</b> directory in the path where the model is located. For details, see the <b>source_location</b> parameter. The script name is fixed to <b>customize_service.py</b> . |
| runtime         | No        | String              | Supported runtime environment. This parameter is mandatory if <b>model_type</b> is used. For details, see .                                                                                                                                                                                                                                                                |
| input_params    | No        | <b>params</b> array | List of input parameters for model inference. By default, this parameter is left blank. If the <b>apis</b> information has been configured in the model configuration file, you do not need to set this parameter. The backend automatically reads the input parameters from the <b>apis</b> field in the configuration file.                                              |
| output_params   | No        | <b>params</b> array | List of output parameters for model inference. By default, this parameter is left blank. If the <b>apis</b> information has been configured in the model configuration file, you do not need to set this parameter. The backend automatically reads the output parameters from the <b>apis</b> field in the configuration file.                                            |



| Parameter    | Mandatory | Type             | Description                                                                                                                                                                                                                                                                                                                                                             |
|--------------|-----------|------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| dependencies | No        | dependency array | Dependency package required for running the code and model. By default, this parameter is left blank. If the <b>dependencies</b> information has been configured in the model configuration file, you do not need to set this parameter. The backend automatically reads the dependencies to be installed from the <b>dependencies</b> field in the configuration file. |
| apis         | No        | String           | List of inference APIs provided by a model. By default, this parameter is left blank. If the <b>apis</b> information has been configured in the model configuration file, you do not need to set this parameter. The backend automatically reads the configured inference API information from the <b>apis</b> field in the configuration file.                         |

Table 9-3 params parameters

| Parameter  | Mandatory | Type   | Description                                                                                                                                        |
|------------|-----------|--------|----------------------------------------------------------------------------------------------------------------------------------------------------|
| url        | Yes       | String | Request path of a model inference API                                                                                                              |
| param_name | Yes       | String | Parameter name, which contains a maximum of 64 characters                                                                                          |
| param_type | Yes       | String | Basic parameter types of JSON schema, including <b>string</b> , <b>object</b> , <b>array</b> , <b>boolean</b> , <b>number</b> , and <b>integer</b> |
| min        | No        | Double | This parameter is optional when <b>param_type</b> is set to <b>int</b> or <b>float</b> . By default, this parameter is left blank.                 |
| max        | No        | Double | This parameter is optional when <b>param_type</b> is set to <b>int</b> or <b>float</b> . By default, this parameter is left blank.                 |
| param_desc | No        | String | Parameter description, which contains a maximum of 100 characters. By default, this parameter is left blank.                                       |

**Table 9-4 dependency** parameters

| Parameter | Mandatory | Type                 | Description                                      |
|-----------|-----------|----------------------|--------------------------------------------------|
| installer | Yes       | String               | Installation mode. Only <b>pip</b> is supported. |
| packages  | Yes       | <b>package</b> array | Collection of dependency packages                |

**Table 9-5 package** parameters

| Parameter       | Mandatory | Type   | Description                                                                                                                                                                                                                                                                                                                                                |
|-----------------|-----------|--------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| package_name    | Yes       | String | Name of a dependency package                                                                                                                                                                                                                                                                                                                               |
| package_version | No        | String | Version of a dependency package                                                                                                                                                                                                                                                                                                                            |
| restraint       | No        | String | Version filtering condition. This parameter is mandatory only when <b>package_version</b> exists. Possible values are as follows: <ul style="list-style-type: none"> <li>• <b>EXACT</b>: the specified version</li> <li>• <b>ATLEAST</b>: not earlier than the specified version</li> <li>• <b>ATMOST</b>: not later than the specified version</li> </ul> |

**Table 9-6 create\_model** response parameters

| Parameter      | Mandatory | Type         | Description                                                          |
|----------------|-----------|--------------|----------------------------------------------------------------------|
| model_instance | Yes       | Model object | Model object, which can be any of the APIs described in this chapter |

 **NOTE**

Example of creating a model in a handwritten digit recognition project using MXNet:

```
from modelarts.session import Session
from modelarts.model import Model

session = Session()
model_instance = Model(session,
 model_name="digit_recognition",
 model_version="1.0.0",
 source_location=model_location,
 model_type="MXNet",
 model_algorithm="image_classification"
)
```

## 9.2 Obtaining Models

### Sample Code

In ModelArts notebook, you do not need to enter authentication parameters for session authentication. For details about session authentication of other development environments, see [Session Authentication](#).

- **Scenario 1:** Obtain all models of a user.

```
from modelarts.session import Session
from modelarts.model import Model

session = Session()
model_list = Model.get_model_list(session)
```

- **Scenario 2:** Obtain the models of a user based on search criteria.

```
from modelarts.session import Session
from modelarts.model import Model

session = Session()
model_list = Model.get_model_list(session, model_status="published", model_name="digit",
order="desc")
```

### Parameters

**Table 9-7** Query parameters

| Parameter      | Mandatory | Type    | Description                                                                                                                           |
|----------------|-----------|---------|---------------------------------------------------------------------------------------------------------------------------------------|
| model_name     | No        | String  | Model name. Fuzzy match is supported.                                                                                                 |
| model_version  | No        | String  | Model version                                                                                                                         |
| model_statuses | No        | String  | Model status. The value can be <b>publishing</b> , <b>published</b> , or <b>failed</b> . You can obtain jobs based on their statuses. |
| description    | No        | String  | Description. Fuzzy match is supported.                                                                                                |
| offset         | No        | Integer | Index of the page to be queried. Default value: <b>0</b>                                                                              |

| Parameter    | Mandatory | Type    | Description                                                                                                                          |
|--------------|-----------|---------|--------------------------------------------------------------------------------------------------------------------------------------|
| limit        | No        | Integer | Maximum number of records returned on each page. Default value: <b>280</b>                                                           |
| sort_by      | No        | String  | Sorting mode. The value can be <b>create_at</b> , <b>model_version</b> , or <b>model_size</b> . Default value: <b>create_at</b>      |
| order        | No        | String  | Sorting order. The value can be <b>asc</b> or <b>desc</b> , indicating the ascending or descending order. Default value: <b>desc</b> |
| workspace_id | No        | String  | Workspace ID. Default value: <b>0</b>                                                                                                |

**Table 9-8** get\_model\_list parameters

| Parameter   | Type               | Description                                                                        |
|-------------|--------------------|------------------------------------------------------------------------------------|
| total_count | Integer            | Total number of models that meet the search criteria when no paging is implemented |
| count       | Integer            | Number of models                                                                   |
| models      | <b>model</b> array | Model metadata                                                                     |

**Table 9-9** model parameters

| Parameter     | Type   | Description                                                                                                                                                                       |
|---------------|--------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| model_id      | String | Model ID                                                                                                                                                                          |
| model_name    | String | Model name                                                                                                                                                                        |
| model_version | String | Model version                                                                                                                                                                     |
| model_type    | String | Model type. The value can be <b>TensorFlow</b> , <b>MXNet</b> , <b>Spark_Mllib</b> , <b>Scikit_Learn</b> , <b>XGBoost</b> , <b>MindSpore</b> , <b>Image</b> , or <b>PyTorch</b> . |
| model_size    | Long   | Model size, in bytes                                                                                                                                                              |
| tenant        | String | Tenant to whom a model belongs                                                                                                                                                    |
| project       | String | Project to which a model belongs                                                                                                                                                  |
| owner         | String | User to whom a model belongs                                                                                                                                                      |

| Parameter   | Type   | Description                                                                                                          |
|-------------|--------|----------------------------------------------------------------------------------------------------------------------|
| create_at   | Long   | Time when a model is created, in milliseconds calculated from 1970.1.1 0:0:0 UTC                                     |
| description | String | Model description                                                                                                    |
| source_type | String | Model source type. This parameter is valid only when the model is deployed through ExeML. The value is <b>auto</b> . |

## 9.3 Obtaining Model Objects

### Sample Code

In ModelArts notebook, you do not need to enter authentication parameters for session authentication. For details about session authentication of other development environments, see [Session Authentication](#).

- **Scenario 1:** Obtain all model objects of a user.

```
from modelarts.session import Session
from modelarts.model import Model

session = Session()
model_object_list = Model.get_model_object_list(session)
print(model_object_list)
```

- **Scenario 2:** Obtain the model objects of a user based on search criteria.

```
from modelarts.session import Session
from modelarts.model import Model

session = Session()
model_object_list = Model.get_model_object_list(session, model_status="published",
model_name="digit", order="desc")
print(model_object_list)
```

### Parameters

- You can use this API to obtain the model list. The size of the list is equal to the number of models that have been deployed by the current user. Each element in the list is a model object. The object attributes are the same as those in [Obtaining Details About a Model](#). For example, in `model_list = [model_instance1, model_instance2, model_instance3 ...]`, each `model_instance` in the list is a model API that can be called.
- The model list can be obtained based on the query parameters. [Table 9-10](#) describes the query parameters.
- When the model list is queried, details about the models are returned. See [Table 9-11](#) and [Table 9-12](#).
- A maximum of 150 model objects can be obtained.

**Table 9-10** Query parameters

| Parameter     | Mandatory | Type    | Description                                                                                                                           |
|---------------|-----------|---------|---------------------------------------------------------------------------------------------------------------------------------------|
| model_name    | No        | String  | Model name. Fuzzy match is supported.                                                                                                 |
| model_version | No        | String  | Model version                                                                                                                         |
| model_status  | No        | String  | Model status. The value can be <b>publishing</b> , <b>published</b> , or <b>failed</b> . You can obtain jobs based on their statuses. |
| description   | No        | String  | Description. Fuzzy match is supported.                                                                                                |
| offset        | No        | Integer | Index of the page to be queried. Default value: <b>0</b>                                                                              |
| limit         | No        | Integer | Maximum number of records returned on each page. Default value: <b>280</b>                                                            |
| sort_by       | No        | String  | Sorting mode. The value can be <b>create_at</b> , <b>model_version</b> , or <b>model_size</b> . Default value: <b>create_at</b>       |
| order         | No        | String  | Sorting order. The value can be <b>asc</b> or <b>desc</b> , indicating the ascending or descending order. Default value: <b>desc</b>  |
| workspace_id  | No        | String  | Workspace ID. Default value: <b>0</b>                                                                                                 |

**Table 9-11** get\_model\_list parameters

| Parameter   | Type               | Description                                                                        |
|-------------|--------------------|------------------------------------------------------------------------------------|
| total_count | Integer            | Total number of models that meet the search criteria when no paging is implemented |
| count       | Integer            | Number of models                                                                   |
| models      | <b>model</b> array | Model metadata                                                                     |

**Table 9-12** model parameters

| Parameter | Type   | Description |
|-----------|--------|-------------|
| model_id  | String | Model ID    |

| Parameter     | Type   | Description                                                                                                                                                                       |
|---------------|--------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| model_name    | String | Model name                                                                                                                                                                        |
| model_version | String | Model version                                                                                                                                                                     |
| model_type    | String | Model type. The value can be <b>TensorFlow</b> , <b>MXNet</b> , <b>Spark_MLlib</b> , <b>Scikit_Learn</b> , <b>XGBoost</b> , <b>MindSpore</b> , <b>Image</b> , or <b>PyTorch</b> . |
| model_size    | Long   | Model size, in bytes                                                                                                                                                              |
| tenant        | String | Tenant to whom a model belongs                                                                                                                                                    |
| project       | String | Project to which a model belongs                                                                                                                                                  |
| owner         | String | User to which a model belongs                                                                                                                                                     |
| create_at     | Long   | Time when a model is created, in milliseconds calculated from 1970.1.1 0:0:0 UTC                                                                                                  |
| description   | String | Model description                                                                                                                                                                 |
| source_type   | String | Model source type. This parameter is valid only when the model is deployed by an ExeML project. The value is <b>auto</b> .                                                        |

## 9.4 Obtaining Details About a Model

You can use the API to obtain the information about a model object.

### Sample Code

In ModelArts notebook, you do not need to enter authentication parameters for session authentication. For details about session authentication of other development environments, see [Session Authentication](#).

- Method 1:** Obtain details about a model based on the model object created in [Importing a Model](#).
 

```
from modelarts.session import Session
from modelarts.model import Model

session = Session()
model_instance = Model(session, model_id="your_model_id")
model_info = model_instance.get_model_info()
print(model_info)
```
- Method 2:** Obtain details about a model based on the model object returned in [Obtaining Model Objects](#).
 

```
from modelarts.session import Session
from modelarts.model import Model

session = Session()
model_object_list = Model.get_model_object_list(session)
model_instance = model_object_list[0]
```

```
model_info = model_instance.get_model_info()
print(model_info)
```

## Parameters

**Table 9-13** get\_model\_info response parameters

| Parameter          | Type   | Description                                                                                                                                                                       |
|--------------------|--------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| model_id           | String | Model ID                                                                                                                                                                          |
| model_name         | String | Model name                                                                                                                                                                        |
| model_version      | String | Model version                                                                                                                                                                     |
| tenant             | String | Tenant                                                                                                                                                                            |
| project            | String | Project                                                                                                                                                                           |
| owner              | String | User                                                                                                                                                                              |
| create_at          | Long   | Time when a model is created, in milliseconds calculated from 1970.1.1 0:0:0 UTC                                                                                                  |
| source_location    | String | OBS path where a model resides                                                                                                                                                    |
| source_job_id      | String | ID of the source training job                                                                                                                                                     |
| source_job_version | String | Version of the source training job                                                                                                                                                |
| source_type        | String | Type of a model source <ul style="list-style-type: none"> <li>If a model is deployed by a training job or OBS model file, this parameter is left blank.</li> </ul>                |
| model_type         | String | Model type. The value can be <b>TensorFlow</b> , <b>MXNet</b> , <b>Spark_Mllib</b> , <b>Scikit_Learn</b> , <b>XGBoost</b> , <b>MindSpore</b> , <b>Image</b> , or <b>PyTorch</b> . |
| model_size         | Long   | Model size, in bytes                                                                                                                                                              |
| model_status       | String | Model status. The value can be <b>publishing</b> , <b>published</b> , or <b>failed</b> .                                                                                          |
| description        | String | Model description                                                                                                                                                                 |
| execution_code     | String | OBS path for storing the execution code. The name of the execution code file is fixed to <b>customize_service.py</b> .                                                            |
| schema_doc         | String | Download address of the model schema file                                                                                                                                         |



| Parameter     | Type                    | Description                                                                                                                                                                                        |
|---------------|-------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| image_address | String                  | Execution image path of a model. Before the image is built, that is, before a model has been published as a service, this parameter is left blank.                                                 |
| input_params  | <b>params</b> array     | Collection of input parameters of a model. By default, this parameter is left blank.                                                                                                               |
| output_params | <b>params</b> array     | Collection of output parameters of a model. By default, this parameter is left blank.                                                                                                              |
| dependencies  | <b>dependency</b> array | Package required for running the code and model                                                                                                                                                    |
| model_metrics | String                  | Model evaluation parameter. This parameter is returned only when <b>source_job_id</b> and <b>source_job_version</b> are assigned values and the corresponding training job has evaluation results. |
| apis          | String                  | All <b>apis</b> input and output parameters of the model                                                                                                                                           |

**Table 9-14** params parameters

| Parameter  | Type   | Description                                                                                                                                                                    |
|------------|--------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| url        | String | API URL                                                                                                                                                                        |
| param_name | String | Parameter name, which contains a maximum of 64 characters                                                                                                                      |
| param_type | String | Parameter type. The value can be <b>int</b> , <b>string</b> , <b>float</b> , <b>timestamp</b> , <b>date</b> , or <b>file</b> .                                                 |
| min        | Number | When <b>param_type</b> is set to <b>int</b> or <b>float</b> and <b>min</b> is set during model creation, the value will be returned. By default, this parameter is left blank. |
| max        | Number | When <b>param_type</b> is set to <b>int</b> or <b>float</b> and <b>max</b> is set during model creation, the value will be returned. By default, this parameter is left blank. |
| param_desc | String | Parameter description, which contains a maximum of 100 characters. By default, this parameter is left blank.                                                                   |

**Table 9-15** dependency parameters

| Parameter | Type   | Description |
|-----------|--------|-------------|
| installer | String | Installer   |

| Parameter | Type                    | Description                       |
|-----------|-------------------------|-----------------------------------|
| packages  | <b>package</b><br>array | Collection of dependency packages |

**Table 9-16** package parameters

| Parameter       | Type   | Description                                                                                                                                                                                                                                                                       |
|-----------------|--------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| package_name    | String | Name of a dependency package                                                                                                                                                                                                                                                      |
| package_version | String | Version of a dependency package                                                                                                                                                                                                                                                   |
| restraint       | String | Version filtering criterion. The options are as follows: <ul style="list-style-type: none"> <li>• <b>EXACT</b>: the specified version</li> <li>• <b>ATLEAST</b>: not earlier than the specified version</li> <li>• <b>ATMOST</b>: not later than the specified version</li> </ul> |

**Table 9-17** metric parameters

| Parameter | Mandatory | Type   | Description |
|-----------|-----------|--------|-------------|
| f1        | Yes       | Double | Mean        |
| recall    | Yes       | Double | Recall      |
| precision | Yes       | Double | Precision   |
| accuracy  | Yes       | Double | Accuracy    |

## 9.5 Deleting a Model

You can use the API to delete a model object.

### Sample Code

In ModelArts notebook, you do not need to enter authentication parameters for session authentication. For details about session authentication of other development environments, see [Session Authentication](#).

- **Method 1:** Delete the model object created in [Importing a Model](#).

```
from modelarts.session import Session
from modelarts.model import Model

session = Session()
model_instance = Model(session, model_id="your_model_id")
model_instance.delete_model()
```

- **Method 2:** Delete the model object returned in [Obtaining Model Objects](#).

```
from modelarts.session import Session
from modelarts.model import Model

session = Session()
model_object_list = Model.get_model_object_list(session)
model_instance = model_object_list[0]
model_instance.delete_model()
```

# 10 Service Management

---

## 10.1 Service Management Overview

Service management indicates deploying a model that has been successfully created as a real-time. This feature provides functions such as real-time prediction, service details query, and service log query.

The real-time services include **predictor** and **transformer**, both of which provide the functions described in the following sections. This chapter uses **predictor** as an example.

### NOTE

The sample code in this chapter is implemented in ModelArts notebook instances. If the code is used in other development environments, the session needs to be authenticated. For details about session authentication, see [Session Authentication](#).

## 10.2 Deploying a Real-Time Service

Real-time service deployment covers the following aspects:

- Initialize a real-time service.
- Deploy a real-time service predictor.
- Deploy a batch service transformer.

The service object predictor is returned after deployment. The attributes of the service object include all functions described in this chapter.

### Sample Code

In ModelArts notebook, you do not need to enter authentication parameters for session authentication. For details about session authentication of other development environments, see [Session Authentication](#).

- Method 1: Initialize the predictor that has been deployed as a real-time service.

```
from modelarts.session import Session
from modelarts.model import Predictor
```

```
session = Session()
predictor_instance = Predictor(session, service_id="your_service_id")
```

- Method 2: Deploy a real-time service predictor.

- Deploy the service in a public resource pool.

```
from modelarts.session import Session
from modelarts.model import Model
from modelarts.config.model_config import ServiceConfig, TransformerConfig, Schedule

session = Session()
model_instance = Model(session, model_id='your_model_id')
vpc_id = None # (Optional) ID of the VPC where the real-time service
instance is deployed. This parameter is left blank by default.
subnet_network_id = None # (Optional) Subnet ID. This parameter is left
blank by default.
security_group_id = None # (Optional) Security group. This parameter is left
blank by default.
configs = [ServiceConfig(model_id=model_instance.model_id,
 weight="100",
 instance_count=1,
 specification="modelarts.vm.cpu.2u")] # For details, see specification.
predictor_instance = model_instance.deploy_predictor(
 service_name="service_predictor_name",
 infer_type="real-time",
 vpc_id=vpc_id,
 subnet_network_id=subnet_network_id,
 security_group_id=security_group_id,
 configs=configs, # predictor configuration parameter. For details, see
configs.
 schedule = [Schedule(op_type='stop', time_unit='HOURS', duration=1)] # (Optional)
Specify the runtime duration for a real-time service.
)
```

The **model\_id** parameter specifies the model that is to be deployed as a real-time service. Obtain the value by calling the API described in [Obtaining Models](#) or from the ModelArts management console.

- Deploy the service in a dedicated resource pool.

```
from modelarts.config.model_config import ServiceConfig

configs = [ServiceConfig(model_id=model_instance.model_id, weight="100", instance_count=1,
 specification="modelarts.vm.cpu.2u")]
predictor_instance = model_instance.deploy_predictor(
 service_name="your_service_name",
 infer_type="real-time",
 configs=configs,
 cluster_id="your dedicated pool id"
)
```

**configs** is defined by **ServiceConfig** in the SDK. The type of **configs** is list, and the tuple object in the list is **ServiceConfig**. The code is as follows:

```
configs = []
envs = {"model_name": "mxnet-model-1", "load_epoch": "0"}

service_config1 = ServiceConfig(
 model_id="model_id1", # model_id1 and model_id2 must be the IDs of different
versions of the same model.
 weight="70",
 specification="modelarts.vm.cpu.2u", # For details, see specification.
 instance_count=2,
 envs=envs) # (Optional) Configure the environment variable, for example,
envs = {"model_name": "mxnet-model-1", "load_epoch": "0"}.
service_config2 = ServiceConfig(
 model_id="model_id2",
 weight="30",
 specification="modelarts.vm.cpu.2u", # For details, see specification.
 instance_count=2,
 envs=envs) # (Optional) Configure the environment variable, for example,
```

```
envs = {"model_name": "mxnet-model-1", "load_epoch": "0"}.
configs.append(service_config1)
configs.append(service_config2)
```

- Method 3: Deploy a batch service transformer.

```
from modelarts.session import Session
from modelarts.model import Model
from modelarts.config.model_config import TransformerConfig

session = Session()
model_instance = Model(session, model_id='your_model_id')
vpc_id = None # (Optional) ID of the VPC where the batch service instance
is deployed. This parameter is left blank by default.
subnet_network_id = None # (Optional) Subnet ID. This parameter is left blank by
default.
security_group_id = None # (Optional) Security group. This parameter is left blank
by default.

transformer = model_instance.deploy_transformer(
 service_name="service_transformer_name",
 infer_type="batch",
 vpc_id=vpc_id,
 subnet_network_id=subnet_network_id,
 security_group_id=security_group_id,
 configs=configs # transformer configuration parameter. For details, see configs.
)
```

**configs** is defined by **TransformerConfig** in the SDK. The type of **configs** is list, and the tuple object in the list is **TransformerConfig**. The code is as follows:

```
configs = []
mapping_rule = None # (Optional) Mapping between input parameters and CSV
data
mapping_type= "file" # File or CSV
envs = {"model_name": "mxnet-model-1", "load_epoch": "0"}

transformer_config1 = TransformerConfig(
 model_id="model_id",
 specification="modelarts.vm.cpu.2u", # For details, see specification.
 instance_count=2,
 src_path="/shp-cn4/sdk-demo/", # OBS path to the input of the batch task, for
example, /your_obs_bucket/src_path
 dest_path="/shp-cn4/data-out/", # OBS path to the output of the batch task, for
example, /your_obs_bucket/dest_path
 req_uri="/",
 mapping_type=mapping_type,
 mapping_rule=mapping_rule,
 envs=envs) # (Optional) Configure the environment variable, for example,
envs = {"model_name": "mxnet-model-1", "load_epoch": "0"}.
configs.append(transformer_config1)
```

## Parameters

Table 10-1 Parameters

| Parameter  | Mandatory | Type   | Description                                                                                      |
|------------|-----------|--------|--------------------------------------------------------------------------------------------------|
| service_id | Yes       | String | Service ID, which can be obtained from the real-time service on the ModelArts management console |

| Parameter | Mandatory | Type   | Description                                                                                               |
|-----------|-----------|--------|-----------------------------------------------------------------------------------------------------------|
| session   | Yes       | Object | Session object. For details about the initialization method, see <a href="#">Session Authentication</a> . |

**Table 10-2** Parameters for deploying the predictor and transformer

| Parameter         | Mandatory | Type   | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |
|-------------------|-----------|--------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| service_name      | No        | String | Name of a service that consists of 1 to 64 characters.                                                                                                                                                                                                                                                                                                                                                                                                                                                               |
| description       | No        | String | Service description, which contains a maximum of 100 characters. By default, this parameter is left blank.                                                                                                                                                                                                                                                                                                                                                                                                           |
| infer_type        | No        | String | Inference mode. The value can be <b>real-time</b> or <b>batch</b> . The default value is <b>real-time</b> . <ul style="list-style-type: none"> <li><b>real-time</b>: real-time service. A model is deployed as a web service and provides real-time test UI and monitoring capabilities. The service keeps running.</li> <li><b>batch</b>: batch service. A batch service can perform inference on batch data and automatically stops after data processing is completed.</li> </ul>                                 |
| vpc_id            | No        | String | ID of the VPC to which a real-time service instance is deployed. By default, this parameter is left blank. In this case, ModelArts allocates a dedicated VPC to each user, and users are isolated from each other. To access other service components in the VPC of the service instance, set this parameter to the ID of the corresponding VPC.<br><br>Once a VPC is configured, it cannot be modified. When <b>vpc_id</b> and <b>cluster_id</b> are configured, only the dedicated cluster parameter takes effect. |
| subnet_network_id | No        | String | ID of a subnet. By default, this parameter is left blank. This parameter is mandatory when <b>vpc_id</b> is configured. Enter the network ID displayed in the subnet details on the VPC management console. A subnet provides dedicated network resources that are isolated from other networks.                                                                                                                                                                                                                     |

| Parameter         | Mandatory | Type                                                                 | Description                                                                                                                                                                                                                                                                                                                                                                                                            |
|-------------------|-----------|----------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| security_group_id | No        | String                                                               | Security group. By default, this parameter is left blank. This parameter is mandatory when <b>vpc_id</b> is configured. A security group is a virtual firewall that provides secure network access control policies for service instances. A security group must contain at least one inbound rule to permit the requests whose protocol is TCP, source address is <b>0.0.0.0/0</b> , and port number is <b>8080</b> . |
| configs           | Yes       | <b>configs</b> parameters of <b>predictor</b> and <b>transformer</b> | Model running configurations <ul style="list-style-type: none"> <li>When <b>infer_type</b> is set to <b>batch</b>, only one model can be configured.</li> <li>When <b>infer_type</b> is set to <b>real-time</b>, you can configure multiple models and assign traffic weights based on service requirements. The version numbers of the models must be different.</li> </ul>                                           |
| schedule          | No        | <b>schedule</b> array                                                | Service scheduling configuration, which can be configured only for real-time services. By default, this parameter is not used. Services run for a long time. For details, see <a href="#">Table 10-6</a> .                                                                                                                                                                                                             |
| cluster_id        | No        | String                                                               | ID of an old-version dedicated resource pool, which is left blank by default. If this parameter is configured, the service will be deployed in the specified old-version dedicated resource pool.                                                                                                                                                                                                                      |
| pool_name         | No        | String                                                               | Name of a new-version dedicated resource pool.                                                                                                                                                                                                                                                                                                                                                                         |

**Table 10-3** configs parameters of **predictor**

| Parameter | Mandatory | Type   | Description                                                                                                                           |
|-----------|-----------|--------|---------------------------------------------------------------------------------------------------------------------------------------|
| model_id  | Yes       | String | Model ID. Obtain the value by calling the API described in <a href="#">Obtaining Models</a> or from the ModelArts management console. |



| Parameter      | Mandatory | Type                | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
|----------------|-----------|---------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| weight         | Yes       | Integer             | <p>Weight of traffic allocated to a model. This parameter is mandatory only when <b>infer_type</b> is set to <b>real-time</b>. The sum of multiple weights must be equal to 100. If multiple model versions are configured in a real-time service and different traffic weights are set, ModelArts continuously accesses the prediction API of the service and forwards prediction requests to the model instances of the corresponding versions based on the weights.</p> <pre> {   "service_name": "mnist",   "description": "mnist service",   "infer_type": "real-time",   "config": [     {       "model_id": "xxxmodel-idxxx",       "weight": "70",       "specification": "modelarts.vm.cpu.2u",       "instance_count": 1,       "envs": {         "model_name": "mxnet-model-1",         "load_epoch": "0"       }     },     {       "model_id": "xxxxxx",       "weight": "30",       "specification": "modelarts.vm.cpu.2u",       "instance_count": 1     }   ] } </pre> |
| specification  | Yes       | String              | Resource specifications.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |
| instance_count | Yes       | Integer             | Number of instances deployed in a model. The maximum number of instances is 5. To use more instances, submit a service ticket.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |
| envs           | No        | Map<String, String> | (Optional) Environment variable key-value pair required for running a model. By default, this parameter is left blank.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |

**Table 10-4** configs parameters of transformer

| Parameter | Mandatory | Type   | Description |
|-----------|-----------|--------|-------------|
| model_id  | Yes       | String | Model ID    |

| Parameter      | Mandatory | Type                | Description                                                                                                                                                                                                                                                                     |
|----------------|-----------|---------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| specification  | Yes       | String              | Resource flavor. Currently, <b>modelarts.vm.cpu.2u</b> and <b>modelarts.vm.gpu.p4</b> are available.                                                                                                                                                                            |
| instance_count | Yes       | Integer             | Number of instances deployed in a model. The value range during the closed beta test is [1, 2].                                                                                                                                                                                 |
| envs           | No        | Map<String, String> | (Optional) Environment variable key-value pair required for running a model. By default, this parameter is left blank.                                                                                                                                                          |
| src_path       | Yes       | String              | OBS path of the input data of a batch job                                                                                                                                                                                                                                       |
| dest_path      | Yes       | String              | OBS path of the output data of a batch job                                                                                                                                                                                                                                      |
| req_uri        | Yes       | String              | Inference API called in a batch task, that is, the RESTful API exposed in the model image. You must select an API URL from the <b>config.json</b> file of the model for inference. If a built-in inference image of ModelArts is used, the API is displayed as <code>/</code> . |

| Parameter    | Mandatory | Type   | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |
|--------------|-----------|--------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| mapping_type | Yes       | String | <p>Mapping type of the input data. The value can be <b>file</b> or <b>csv</b>.</p> <ul style="list-style-type: none"> <li>If you select <b>file</b>, each inference request corresponds to a file in the input data path. When this mode is used, <b>req_uri</b> of a model can have only one input parameter and the type of this parameter is <b>file</b>.</li> <li>If you select <b>csv</b>, each inference request corresponds to a row of data in the CSV file. When this mode is used, the files in the input data path can only be in CSV format and <b>mapping_rule</b> needs to be configured to map the index of each parameter in the inference request body to the CSV file.</li> </ul> <p>The following shows how to create a batch service whose <b>mapping_type</b> is set to <b>file</b>:</p> <pre>{   "service_name": "batchservicetest",   "description": "",   "infer_type": "batch",   "config": [{     "model_id": "598b913a-af3e-41ba-a1b5-bf065320f1e2",     "specification": "modelarts.vm.cpu.2u",     "instance_count": 1,     "src_path": "https://infers-data.obs.xxx.com/xgboosterdata/",     "dest_path": "https://infers-data.obs.xxx.com/output/",     "req_uri": "/",     "mapping_type": "file"   }] }</pre> <p>The following shows how to create a batch service whose <b>mapping_type</b> is set to <b>csv</b>:</p> <pre>{   "service_name": "batchservicetest",   "description": "",   "infer_type": "batch",   "config": [{     "model_id": "598b913a-af3e-41ba-a1b5-bf065320f1e2",     "specification": "modelarts.vm.cpu.2u",     "instance_count": 1,     "src_path": "https://infers-data.obs.xxx.com/xgboosterdata/",     "dest_path": "https://infers-data.obs.xxx.com/output/",     "req_uri": "/",     "mapping_type": "csv",     "mapping_rule": {       "type": "object",       "properties": {         "data": {           "type": "object",           "properties": {             "req_data": {               "type": "array",               "items": [{                 "type": "object",                 "properties": { </pre> |



**Table 10-5** Parameters in the response to the request for deploying **predictor** and **transformer**

| Parameter | Mandatory | Type             | Description                                                                       |
|-----------|-----------|------------------|-----------------------------------------------------------------------------------|
| predictor | Yes       | Predictor object | Predictor object. Its attributes include all functions described in this chapter. |

**Table 10-6** **schedule** parameters

| Parameter | Mandatory | Type    | Description                                                                                                                                              |
|-----------|-----------|---------|----------------------------------------------------------------------------------------------------------------------------------------------------------|
| op_type   | Yes       | String  | Scheduling type. Currently, only the value <b>stop</b> is supported.                                                                                     |
| time_unit | Yes       | String  | Scheduling time unit. The options are as follows: <ul style="list-style-type: none"> <li>• DAYS</li> <li>• HOURS</li> <li>• MINUTES</li> </ul>           |
| duration  | Yes       | Integer | Value that maps to the time unit. For example, if the task stops after two hours, set <b>time_unit</b> to <b>HOURS</b> and <b>duration</b> to <b>2</b> . |

 NOTE

- Example of deploying a real-time **predictor** instance in the handwritten digit recognition project implemented by MXNet:

```
from modelarts.session import Session
from modelarts.model import Model
from modelarts.config.model_config import ServiceConfig, TransformerConfig

model_instance = Model(session, model_id = "you_model_id")
configs = []
config1 = ServiceConfig(model_id="you_model_id",
 weight="100",
 instance_count=1,
 specification="modelarts.vm.cpu.2u",
 envs={"input_data_name":"images",
 "input_data_shape":"0,1,28,28",
 "output_data_shape":"0,10"})
configs.append(config1)
predictor = model_instance.deploy_predictor(service_name="DigitRecognition", configs=configs)
```

- Example of deploying a **transformer** instance (batch processing) in a handwritten digit recognition project implemented by MXNet:

```
from modelarts.session import Session
from modelarts.model import Model
from modelarts.config.model_config import ServiceConfig, TransformerConfig

model_instance = Model(session, model_id = "your_model_id")
configs = []
config1 = TransformerConfig(model_id="your_model_id",
 specification="modelarts.vm.cpu.2u",
 instance_count=1,

envs={"input_data_name":"images","input_data_shape":"0,1,28,28","output_data_shape":"0,10"},
 src_path="/w0403/testdigitrecognition/inferimages/",
 dest_path="/w0403/testdigitrecognition/" ,
 req_uri = "/",
 mapping_type = "file")
configs.append(config1)
predictor = model_instance.deploy_transformer(service_name="DigitRecognition",
infer_type="batch", configs=configs)
```

## 10.3 Obtaining Details About a Service

You can use the API to obtain details about a service object.

### Sample Code

In ModelArts notebook, you do not need to enter authentication parameters for session authentication. For details about session authentication of other development environments, see [Session Authentication](#).

- **Method 1:** Obtain details about a service object created in [Deploying a Real-Time Service](#).

```
from modelarts.session import Session
from modelarts.model import Predictor

session = Session()
predictor_instance = Predictor(session, service_id="your_service_id")
predictor_info = predictor_instance.get_service_info()
print(predictor_info)
```

- **Method 2:** Obtain details about a service based on the service object returned in [Obtaining Service Objects](#).

```
from modelarts.session import Session
from modelarts.model import Predictor
```

```
session = Session()
predictor_object_list = Predictor.get_service_object_list(session)
predictor_instance = predictor_object_list[0]
predictor_info = predictor_instance.get_service_info()
print(predictor_info)
```

## Parameters

**Table 10-7** get\_service\_info response parameters

| Parameter         | Type                                                   | Description                                                                                                                                                                               |
|-------------------|--------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| service_id        | String                                                 | Service ID                                                                                                                                                                                |
| service_name      | String                                                 | Service name                                                                                                                                                                              |
| description       | String                                                 | Service description                                                                                                                                                                       |
| tenant            | String                                                 | Tenant to whom a service belongs                                                                                                                                                          |
| project           | String                                                 | Project to which a service belongs                                                                                                                                                        |
| owner             | String                                                 | User to whom a service belongs                                                                                                                                                            |
| publish_at        | Number                                                 | Latest service publishing time, in milliseconds calculated from 1970.1.1 0:0:0 UTC                                                                                                        |
| infer_type        | String                                                 | Inference mode. The value can be <b>real-time</b> or <b>batch</b> .                                                                                                                       |
| vpc_id            | String                                                 | ID of the VPC to which a service instance belongs. This parameter is returned when the network configuration is customized.                                                               |
| subnet_network_id | String                                                 | ID of the subnet where a service instance resides. This parameter is returned when the network configuration is customized.                                                               |
| security_group_id | String                                                 | Security group to which a service instance belongs. This parameter is returned when the network configuration is customized.                                                              |
| status            | String                                                 | Service status. The value can be <b>running</b> , <b>deploying</b> , <b>concerning</b> , <b>failed</b> , <b>stopped</b> , or <b>finished</b> .                                            |
| error_msg         | String                                                 | Error message. When <b>status</b> is <b>failed</b> , the deployment failure cause is returned.                                                                                            |
| config            | <b>config</b> array corresponding to <b>infer_type</b> | <b>config</b> array corresponding to <b>infer_type</b> Service configurations (If a service is shared, only <b>model_id</b> , <b>model_name</b> , and <b>model_version</b> are returned.) |
| access_addresses  | String                                                 | Access address of an inference request. This parameter is returned when <b>infer_type</b> is set to <b>real-time</b> .                                                                    |

| Parameter        | Type    | Description                                                                              |
|------------------|---------|------------------------------------------------------------------------------------------|
| invocation_times | Number  | Total number of service calls                                                            |
| failed_times     | Number  | Number of failed service calls                                                           |
| is_shared        | Boolean | Whether a service is subscribed                                                          |
| shared_count     | Number  | Number of subscriptions                                                                  |
| progress         | Integer | Deployment progress. This parameter is returned when <b>status</b> is <b>deploying</b> . |

**Table 10-8** config parameters corresponding to **real-time**

| Parameter      | Type                | Description                                                                                                                                                                                                                                                                                                                                        |
|----------------|---------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| model_id       | String              | Model ID. You can obtain the value by calling the API described in <a href="#">Obtaining Models</a> or from the ModelArts management console.                                                                                                                                                                                                      |
| model_name     | String              | Model name                                                                                                                                                                                                                                                                                                                                         |
| model_version  | String              | Model version                                                                                                                                                                                                                                                                                                                                      |
| source_type    | String              | Model source.                                                                                                                                                                                                                                                                                                                                      |
| status         | String              | Running status of a model instance. Possible values are as follows: <ul style="list-style-type: none"> <li>● <b>ready</b>: ready (All instances have been started.)</li> <li>● <b>concerning</b>: partially ready (Some instances are started but some are not.)</li> <li>● <b>notReady</b>: not ready (All instances are not started.)</li> </ul> |
| weight         | Integer             | Traffic weight allocated to a model                                                                                                                                                                                                                                                                                                                |
| specification  | String              | Resource flavor. The value can be <b>modelarts.vm.cpu.2u</b> , <b>modelarts.vm.gpu.p4</b> , or <b>modelarts.vm.ai1.a310</b> .                                                                                                                                                                                                                      |
| envs           | Map<String, String> | Environment variable key-value pair required for running a model                                                                                                                                                                                                                                                                                   |
| instance_count | Integer             | Number of instances deployed in a model                                                                                                                                                                                                                                                                                                            |
| scaling        | Boolean             | Whether auto scaling is enabled                                                                                                                                                                                                                                                                                                                    |



**Table 10-9 config** parameters corresponding to **batch**

| Parameter      | Type                | Description                                                                                                                                   |
|----------------|---------------------|-----------------------------------------------------------------------------------------------------------------------------------------------|
| model_id       | String              | Model ID. You can obtain the value by calling the API described in <a href="#">Obtaining Models</a> or from the ModelArts management console. |
| model_name     | String              | Model name                                                                                                                                    |
| model_version  | String              | Model version                                                                                                                                 |
| specification  | String              | Resource flavor. The value can be <b>modelarts.vm.cpu.2u</b> or <b>modelarts.vm.gpu.p4</b> .                                                  |
| envs           | Map<String, String> | Environment variable key-value pair required for running a model                                                                              |
| instance_count | Integer             | Number of instances deployed in a model                                                                                                       |
| src_path       | String              | OBS path of the input data of a batch job                                                                                                     |
| dest_path      | String              | OBS path of the output data of a batch job                                                                                                    |
| req_uri        | String              | Inference path of a batch job                                                                                                                 |
| mapping_type   | String              | Mapping type of the input data. The value can be <b>file</b> or <b>csv</b> .                                                                  |
| mapping_rule   | Map                 | Mapping between input parameters and CSV data. This parameter is returned only when <b>mapping_type</b> is set to <b>csv</b> .                |

## 10.4 Obtaining Services

Obtain the service list of a user.

### Sample Code

In ModelArts notebook, you do not need to enter authentication parameters for session authentication. For details about session authentication of other development environments, see [Session Authentication](#).

- Scenario 1:** Obtain all services of a user.
 

```
from modelarts.session import Session
from modelarts.model import Predictor

session = Session()
predictor_list = Predictor.get_service_list(session)
print(predictor_list)
```
- Scenario 2:** Obtain the services of a user based on search criteria.
 

```
from modelarts.session import Session
from modelarts.model import Predictor

session = Session()
predictor_list = Predictor.get_service_list(session, service_name="digit", order="asc", offset="0",
```

```
infer_type="real-time")
print(predictor_list)
```

## Parameters

**Table 10-10** Query parameters

| Parameter      | Mandatory | Type    | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
|----------------|-----------|---------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| session        | Yes       | Object  | Session object. For details about the initialization method, see <a href="#">Session Authentication</a> .                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |
| service_id     | No        | String  | Service ID. By default, the service ID is not filtered.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
| service_name   | No        | String  | Service name. By default, the service name is not filtered.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
| infer_type     | No        | String  | Inference mode. The value can be <b>real-time</b> or <b>batch</b> . By default, this parameter is left blank.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
| offset         | No        | Integer | Start page of the paging list. Default value: <b>0</b>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |
| limit          | No        | Integer | Maximum number of records returned on each page. Default value: <b>1000</b>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
| service_status | No        | String  | Service status. By default, the service status is not filtered. The service list can be queried based on the service status. Possible values are as follows: <ul style="list-style-type: none"> <li>• <b>running</b>: The service is running properly and is being billed.</li> <li>• <b>deploying</b>: The service is being deployed or scheduling resources are being deployed.</li> <li>• <b>concerning</b>: An alarm is generated, indicating that the backend instance is abnormal and may be billed. For example, in the case of multiple instances, some instances are normal, but some are not. A normal instance is billed but is in the <b>concerning</b> status.</li> <li>• <b>failed</b>: The service fails to be deployed. For details about the failure cause, see the event and log.</li> <li>• <b>stopped</b>: The service has been stopped.</li> <li>• <b>finished</b>: This status is displayed only for the batch service, indicating that the service running is completed.</li> </ul> |

| Parameter | Mandatory | Type   | Description                                                                                                                          |
|-----------|-----------|--------|--------------------------------------------------------------------------------------------------------------------------------------|
| sort_by   | No        | String | Sorting mode. The value can be <b>publish_at</b> or <b>service_name</b> . Default value: <b>publish_at</b>                           |
| order     | No        | String | Sorting order. The value can be <b>asc</b> or <b>desc</b> , indicating the ascending or descending order. Default value: <b>desc</b> |
| model_id  | No        | String | Model ID. By default, the model ID is not filtered.                                                                                  |

**Table 10-11** get\_service\_list response parameters

| Parameter   | Type                 | Description                                                                                                                                      |
|-------------|----------------------|--------------------------------------------------------------------------------------------------------------------------------------------------|
| total_count | Integer              | Total number of services that meet the search criteria when no paging is implemented                                                             |
| count       | Integer              | Number of services in the query result. If <b>offset</b> and <b>limit</b> are not set, the values of <b>count</b> and <b>total</b> are the same. |
| services    | <b>service</b> array | Collection of the queried services                                                                                                               |

**Table 10-12** service parameters

| Parameter    | Type   | Description                                                                                                                                    |
|--------------|--------|------------------------------------------------------------------------------------------------------------------------------------------------|
| service_id   | String | Service ID                                                                                                                                     |
| service_name | String | Service name                                                                                                                                   |
| description  | String | Service description                                                                                                                            |
| tenant       | String | Tenant to whom a service belongs                                                                                                               |
| project      | String | Project to which a service belongs                                                                                                             |
| owner        | String | User to whom a service belongs                                                                                                                 |
| publish_at   | Number | Latest service publishing time, in milliseconds calculated from 1970.1.1 0:0:0 UTC                                                             |
| infer_type   | String | Inference mode. The value can be <b>real-time</b> or <b>batch</b> .                                                                            |
| status       | String | Service status. The value can be <b>running</b> , <b>deploying</b> , <b>concerning</b> , <b>failed</b> , <b>stopped</b> , or <b>finished</b> . |

| Parameter        | Type    | Description                                                                              |
|------------------|---------|------------------------------------------------------------------------------------------|
| progress         | Integer | Deployment progress. This parameter is returned when <b>status</b> is <b>deploying</b> . |
| invocation_times | Number  | Total number of service calls                                                            |
| failed_times     | Number  | Number of failed service calls                                                           |
| is_shared        | Boolean | Whether a service is subscribed                                                          |
| shared_count     | Number  | Number of subscriptions                                                                  |

## 10.5 Obtaining Service Objects

You can use the API to obtain the service object list of a user.

### Sample Code

In ModelArts notebook, you do not need to enter authentication parameters for session authentication. For details about session authentication of other development environments, see [Session Authentication](#).

- **Scenario 1:** Obtain all service objects of a user.

```
from modelarts.session import Session
from modelarts.model import Predictor

session = Session()
predictor_list_object_resp = Predictor.get_service_object_list(session)
print(predictor_list_object_resp)
```

- **Scenario 2:** Obtain the service objects of a user based on search criteria.

```
from modelarts.session import Session
from modelarts.model import Predictor

session = Session()
predictor_object_list = Predictor.get_service_object_list(session, service_name="digit", order="asc",
offset="0", infer_type="real-time")
print(predictor_object_list)
```

### Parameters

- You can use the API to obtain the service list. The list size is equal to the number of services deployed by the user. Each element in the list is a predictor object. The object attributes are the same as those in service initialization.  
For example, in **service\_list\_resp = [service\_instance1, service\_instance2, service\_instance3 ...]**, each **service\_instance** in the list is a service API that can be called in the service management section.
- The service list can be queried based on the query parameters. [Table 10-13](#) describes the query parameters.
- When the model list is queried, details about the services are returned. See [Table 10-14](#) and [Table 10-15](#).

**Table 10-13** Query parameters

| Parameter    | Mandatory | Type    | Description                                                                                                                          |
|--------------|-----------|---------|--------------------------------------------------------------------------------------------------------------------------------------|
| session      | Yes       | Object  | Session object. For details about the initialization method, see <a href="#">Session Authentication</a> .                            |
| is_show      | No        | Boolean | Whether to print service object information. Default value: <b>True</b>                                                              |
| service_id   | No        | String  | Service ID. By default, the service ID is not filtered.                                                                              |
| service_name | No        | String  | Service name. By default, the service name is not filtered.                                                                          |
| infer_type   | No        | String  | Inference mode. The value can be <b>real-time</b> or <b>batch</b> . By default, this parameter is left blank.                        |
| offset       | No        | Integer | Start page of the paging list. Default value: <b>0</b>                                                                               |
| limit        | No        | Integer | Maximum number of records returned on each page. Default value: <b>1000</b>                                                          |
| sort_by      | No        | String  | Sorting mode. The value can be <b>publish_at</b> or <b>service_name</b> . Default value: <b>publish_at</b>                           |
| order        | No        | String  | Sorting order. The value can be <b>asc</b> or <b>desc</b> , indicating the ascending or descending order. Default value: <b>desc</b> |
| model_id     | No        | String  | Model ID. By default, the model ID is not filtered.                                                                                  |

**Table 10-14** `get_service_list` response parameters

| Parameter   | Type                 | Description                                                                                                                                      |
|-------------|----------------------|--------------------------------------------------------------------------------------------------------------------------------------------------|
| total_count | Integer              | Total number of services that meet the search criteria when no paging is implemented                                                             |
| count       | Integer              | Number of services in the query result. If <b>offset</b> and <b>limit</b> are not set, the values of <b>count</b> and <b>total</b> are the same. |
| services    | <b>service</b> array | Collection of the queried services                                                                                                               |

**Table 10-15** service parameters

| Parameter        | Type    | Description                                                                                                                                    |
|------------------|---------|------------------------------------------------------------------------------------------------------------------------------------------------|
| service_id       | String  | Service ID                                                                                                                                     |
| service_name     | String  | Service name                                                                                                                                   |
| description      | String  | Service description                                                                                                                            |
| tenant           | String  | Tenant to whom a service belongs                                                                                                               |
| project          | String  | Project to which a service belongs                                                                                                             |
| owner            | String  | User to whom a service belongs                                                                                                                 |
| publish_at       | Number  | Latest service publishing time, in milliseconds calculated from 1970.1.1 0:0:0 UTC                                                             |
| infer_type       | String  | Inference mode. The value can be <b>real-time</b> or <b>batch</b> .                                                                            |
| status           | String  | Service status. The value can be <b>running</b> , <b>deploying</b> , <b>concerning</b> , <b>failed</b> , <b>stopped</b> , or <b>finished</b> . |
| progress         | Integer | Deployment progress. This parameter is returned when <b>status</b> is <b>deploying</b> .                                                       |
| invocation_times | Number  | Total number of service calls                                                                                                                  |
| failed_times     | Number  | Number of failed service calls                                                                                                                 |
| is_shared        | Boolean | Whether a service is subscribed                                                                                                                |
| shared_count     | Number  | Number of subscriptions                                                                                                                        |

## 10.6 Updating Service Configurations

You can use the API to update the configurations of a service object.

### Sample Code

In ModelArts notebook, you do not need to enter authentication parameters for session authentication. For details about session authentication of other development environments, see [Session Authentication](#).

- **Method 1:** Update the configurations of a service based on the service object created in [Deploying a Real-Time Service](#).

```
from modelarts.session import Session
from modelarts.model import Predictor
from modelarts.config.model_config import ServiceConfig

session = Session()
predictor_instance = Predictor(session, service_id="your_service_id")
```

```
configs = [ServiceConfig(weight="100", instance_count=1,
specification="modelarts.vm.cpu.2u",model_id="your_model_id")]
service_config = predictor_instance.update_service_config(description="description",
status="running",
configs=configs)
```

- **Method 2:** Update the configurations of a service based on the service object returned in [Obtaining Service Objects](#).

```
from modelarts.session import Session
from modelarts.model import Predictor
from modelarts.config.model_config import ServiceConfig

session = Session()
predictor_object_list = Predictor.get_service_object_list(session)
predictor_instance = predictor_object_list[0]
configs = [ServiceConfig(weight="100", instance_count=1,
specification="modelarts.vm.cpu.2u",model_id="your_model_id")]
predictor_config = predictor_instance.update_service_config(description="description",
status="running",
configs=configs)
```

## Parameters

**Table 10-16** Parameters for deploying predictor

| Parameter   | Mandatory | Type                                                    | Description                                                                                                                                                                                                                                                                   |
|-------------|-----------|---------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| description | No        | String                                                  | Service description, which contains a maximum of 100 characters. If this parameter is not set, the service description is not updated.                                                                                                                                        |
| status      | No        | String                                                  | Service status. The value can be <b>running</b> or <b>stopped</b> . If this parameter is not set, the service status is not changed. <b>status</b> and <b>configs</b> cannot be modified at the same time. If both parameters exist, modify only the <b>status</b> parameter. |
| configs     | No        | <b>predictor configs</b> and <b>transformer configs</b> | Service configurations. If this parameter is not set, the service is not updated. For details about how to generate <b>configs</b> , see <a href="#">Deploying a Real-Time Service</a> .                                                                                      |

### NOTE

The restrictions on updating service configurations are as follows:

- The specified **status** cannot be the same as the current service status.
- If the service status is **deploying**, **stopping**, or **deleting**, **status** cannot be set to **running** or **configs** is not allowed to configure.
- If the service status is **waiting**, **status** cannot be set to **running**.
- If the service status is **concerning**, **status** cannot be set to **running**.

**Table 10-17 configs** parameters of **predictor**

| Parameter       | Mandatory | Type                | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                          |
|-----------------|-----------|---------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| model_id        | Yes       | String              | Model ID. You can obtain the value by calling the API described in <a href="#">Obtaining Models</a> or from the ModelArts management console.                                                                                                                                                                                                                                                                                                                        |
| weight          | Yes       | Integer             | Weight of traffic allocated to a model. This parameter is mandatory only when <b>infer_type</b> is set to <b>real-time</b> . The sum of multiple weights must be equal to 100. If multiple model versions are configured in a real-time service and different traffic weights are set, ModelArts continuously accesses the prediction API of the service and forwards prediction requests to the model instances of the corresponding versions based on the weights. |
| specificat ion  | Yes       | String              | Resource flavor.                                                                                                                                                                                                                                                                                                                                                                                                                                                     |
| instance_ count | Yes       | Integer             | Number of instances deployed in a model. The maximum number of instances is 5. To use more instances, submit a service ticket.                                                                                                                                                                                                                                                                                                                                       |
| envs            | No        | Map<String, String> | (Optional) Environment variable key-value pair required for running a model. By default, this parameter is left blank.                                                                                                                                                                                                                                                                                                                                               |

**Table 10-18 configs** parameters of **transformer**

| Parameter       | Mandatory | Type                | Description                                                                                                                                   |
|-----------------|-----------|---------------------|-----------------------------------------------------------------------------------------------------------------------------------------------|
| model_id        | Yes       | String              | Model ID. You can obtain the value by calling the API described in <a href="#">Obtaining Models</a> or from the ModelArts management console. |
| specificat ion  | Yes       | String              | Resource flavor. Currently, <b>modelarts.vm.cpu.2u</b> and <b>modelarts.vm.gpu.p4</b> are available.                                          |
| instance_ count | Yes       | Integer             | Number of instances deployed in a model. The maximum number of instances is 5. To use more instances, submit a service ticket.                |
| envs            | No        | Map<String, String> | (Optional) Environment variable key-value pair required for running a model. By default, this parameter is left blank.                        |
| src_path        | Yes       | String              | OBS path of the input data of a batch job                                                                                                     |



| Parameter    | Mandatory | Type   | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
|--------------|-----------|--------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| dest_path    | Yes       | String | OBS path of the output data of a batch job                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
| req_uri      | Yes       | String | Inference API called in batch tasks. You must select an API URL from the <b>config.json</b> file of the model for inference.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |
| mapping_type | Yes       | String | Mapping type of the input data. The value can be <b>file</b> or <b>csv</b> . <ul style="list-style-type: none"> <li>If you select <b>file</b>, each inference request corresponds to a file in the input data path. When this mode is used, <b>req_uri</b> of a model can have only one input parameter and the type of this parameter is <b>file</b>.</li> <li>If you select <b>csv</b>, each inference request corresponds to a row of data in the CSV file. When this mode is used, the files in the input data path can only be in CSV format and <b>mapping_rule</b> needs to be configured to map the index of each parameter in the inference request body to the CSV file.</li> </ul> |
| mapping_rule | No        | Map    | Mapping between input parameters and CSV data. This parameter is mandatory only when <b>mapping_type</b> is set to <b>csv</b> . The mapping rule is similar to the definition of the input parameters in the <b>config.json</b> file. You only need to configure the index parameters under each parameter of the string, number, integer, or boolean type, and the value of this parameter to the values of the index parameters in the CSV file to send an inference request. Use commas (,) to separate multiple pieces of CSV data. The values of the index parameters start from <b>0</b> . If the value of the index parameter is <b>-1</b> , ignore this parameter.                    |

**Table 10-19** update\_service\_config response parameters

| Parameter  | Mandatory | Type   | Description                                                                                       |
|------------|-----------|--------|---------------------------------------------------------------------------------------------------|
| error_code | Yes       | String | Error code when the API call fails.<br>This parameter is not included when the API call succeeds. |

| Parameter     | Mandatory | Type   | Description                                                                                          |
|---------------|-----------|--------|------------------------------------------------------------------------------------------------------|
| error_message | Yes       | String | Error message when the API call fails.<br>This parameter is not included when the API call succeeds. |

## 10.7 Obtaining Service Monitoring Information

You can use the API to obtain the monitoring information about a service.

### Sample Code

In ModelArts notebook, you do not need to enter authentication parameters for session authentication. For details about session authentication of other development environments, see [Session Authentication](#).

- **Method 1:** Obtain the monitoring information of a service based on the service object created in [Deploying a Real-Time Service](#).

```
from modelarts.session import Session
from modelarts.model import Predictor

session = Session()
predictor_instance = Predictor(session, service_id="your_service_id")
predictor_monitor = predictor_instance.get_service_monitor()
print(predictor_monitor)
```

- **Method 2:** Obtain the monitoring information of a service based on the service object returned in [Obtaining Service Objects](#).

```
from modelarts.session import Session
from modelarts.model import Predictor

session = Session()
predictor_object_list = Predictor.get_service_object_list(session)
predictor_instance = predictor_object_list[0]
predictor_monitor = predictor_instance.get_service_monitor()
print(predictor_monitor)
```

### Parameters

**Table 10-20** get\_service\_monitor response parameters

| Parameter    | Type                                                                 | Description        |
|--------------|----------------------------------------------------------------------|--------------------|
| service_id   | String                                                               | Service ID         |
| service_name | String                                                               | Service name       |
| monitors     | <b>monitor</b> array corresponding to <b>infer_type</b> of a service | Monitoring details |

**Table 10-21** monitor parameters corresponding to **real-time**

| Parameter        | Type    | Description                           |
|------------------|---------|---------------------------------------|
| model_id         | String  | Model ID                              |
| model_name       | String  | Model name                            |
| model_version    | String  | Model version                         |
| invocation_times | Number  | Total number of model instance calls  |
| failed_times     | Number  | Number of failed model instance calls |
| cpu_core_usage   | Float   | Number of used CPUs                   |
| cpu_core_total   | Float   | Total number of CPUs                  |
| cpu_memory_usage | Integer | Used memory, in MB                    |
| cpu_memory_total | Integer | Total memory, in MB                   |
| gpu_usage        | Float   | Number of used GPUs                   |
| gpu_total        | Float   | Total number of GPUs                  |

## 10.8 Obtaining Service Logs

You can use the API to obtain the logs of a service object.

### Sample Code

In ModelArts notebook, you do not need to enter authentication parameters for session authentication. For details about session authentication of other development environments, see [Session Authentication](#).

- Method 1:** Obtain the logs of a service based on the service object created in [Deploying a Real-Time Service](#).
 

```
from modelarts.session import Session
from modelarts.model import Predictor

session = Session()
predictor_instance = Predictor(session, service_id="your_service_id")
predictor_log = predictor_instance.get_service_logs()
print(predictor_log)
```
- Method 2:** Obtain the logs of a service based on the service object returned in [Obtaining Service Objects](#).
 

```
from modelarts.session import Session
from modelarts.model import Predictor
```

```

session = Session()
predictor_object_list = Predictor.get_service_object_list(session)
predictor_instance = predictor_object_list[0]
predictor_log = predictor_instance.get_service_logs()
print(predictor_log)

```

## Parameters

**Table 10-22** get\_service\_logs response parameters

| Parameter    | Type             | Description         |
|--------------|------------------|---------------------|
| service_id   | String           | Service ID          |
| service_name | String           | Service name        |
| logs         | <b>log</b> array | Service update logs |

**Table 10-23** log parameters

| Parameter   | Type                | Description                                                                                                    |
|-------------|---------------------|----------------------------------------------------------------------------------------------------------------|
| update_time | Long                | Time when a service is updated, in milliseconds calculated from 1970.1.1 0:0:0 UTC                             |
| result      | String              | Update result. The value can be <b>SUCCESS</b> , <b>FAIL</b> , or <b>RUNNING</b> .                             |
| config      | <b>config</b> array | Updated service configurations. This parameter is returned when <b>infer_type</b> is set to <b>real-time</b> . |

**Table 10-24** config parameters

| Parameter      | Type    | Description                             |
|----------------|---------|-----------------------------------------|
| model_id       | String  | Model ID                                |
| model_name     | String  | Model name                              |
| model_version  | String  | Model version                           |
| weight         | Integer | Traffic weight allocated to a model     |
| specification  | String  | Resource flavor                         |
| instance_count | Integer | Number of instances deployed in a model |

| Parameter | Type                | Description                                                      |
|-----------|---------------------|------------------------------------------------------------------|
| envs      | Map<String, String> | Environment variable key-value pair required for running a model |

**Table 10-25** result parameters

| Parameter | Type    | Description                                                                                                    |
|-----------|---------|----------------------------------------------------------------------------------------------------------------|
| node_name | String  | Name of an edge node                                                                                           |
| operation | String  | Operation type. The value can be <b>deploy</b> or <b>delete</b> .                                              |
| result    | Boolean | Operation result. <b>true</b> indicates a successful operation, and <b>false</b> indicates a failed operation. |

## 10.9 Delete a Service

You can delete a service in either of the following ways:

- Delete the service created in [Deploying a Real-Time Service](#).
- Delete the service object returned in [Obtaining Service Objects](#).

### Sample Code

In ModelArts notebook, you do not need to enter authentication parameters for session authentication. For details about session authentication of other development environments, see [Session Authentication](#).

- **Method 1:** Delete a service based on the service object created in [Deploying a Real-Time Service](#).

```
from modelarts.session import Session
from modelarts.model import Predictor

session = Session()
predictor_instance = Predictor(session, service_id="your_service_id")
predictor_instance.delete_service()
```
- **Method 2:** Delete a service based on the service object returned in [Obtaining Service Objects](#).

```
from modelarts.session import Session
from modelarts.model import Predictor

session = Session()
predictor_object_list = Predictor.get_service_object_list(session)
predictor_instance = predictor_object_list[0]
predictor_instance.delete_service()
```